

自動車ソフトウェア要求仕様書の二段階インスペクション方法の 提案と適用評価

蛸島 昭之^{†1} 青山 幹雄^{†2}

概要: 自動車ソフトウェアの規模と複雑度の増加, ならびに安全性や信頼性への高い要求に伴い, ソフトウェア要求仕様書 (SRS) の重要度が増している. しかし, 自動車ソフトウェアの SRS は多様性が高いため, 体系的なインスペクションの適用が困難である. その結果, インスペクションの効果がインスペクタの経験とドメイン知識に依存している. 本稿では, a) ドメイン知識を必要としない第三者インスペクションと b) ドメイン知識を必要とするプロジェクトインスペクションからなる二段階インスペクションを提案する. 提案方法では基準となる参照 SRS と自動車ソフトウェア向けの品質特性を定義し, 第三者インスペクションによって SRS のドキュメント品質を定量的に測定する方法を提案する. 提案方法を 4 つの製品カテゴリから収集した実際の SRS に対して適用し, その有効性を評価した.

キーワード: 要求工学, ソフトウェア要求仕様書, インスペクション, 自動車ソフトウェア

A Method of Two-Stage Inspection for Software Requirements Specifications and Its Evaluation with Multi-Domain Automotive Products

AKIYUKI TAKOSHIMA^{†1} MIKIO AOYAMA^{†2}

Abstract: Automotive software is facing the increase of size and complexity, and strenuous demands for safety and reliability. Software requirements specification (SRS) is consequently gaining importance in automotive software development. However, the diversity in automotive SRSs is becoming an obstacle to apply systematic inspection methods. The performance of an inspection, as a result, largely depends upon the inspector's experience and domain knowledge. This paper proposes a two-stage inspection method, which combines a) first-stage inspection by third-party inspectors, and b) second-stage inspection by the project team. We define a reference SRS and quality characteristics for automotive software in order to propose a third-party inspection method that can quantitatively measure the document quality of SRSs. We applied the proposed method to actual SRSs collected from four product domains, and evaluated its effectiveness.

Keywords: Requirements Engineering, Software Requirements Specification, Inspection, Automotive Software.

1. はじめに

自動車におけるソフトウェアの重要性が増すにつれて, 自動車ソフトウェアの規模は年々増大し, その複雑度は 10 年ごとに 10 倍のペースで増加している [4, 8, 12]. あわせて, 安全性や信頼性に対する厳しい要求もある. このようなソフトウェアの開発を所定の予算と期間を満たしつつ完了するには要求工学 (RE) の導入が不可欠である [11]. なかでもソフトウェア要求仕様書 (SRS) のインスペクションがソフトウェアの品質, 開発コスト, 期間を満たす上で効果があることが知られている [19]. 自動車産業は伝統的に垂直統合型の構造をとってきたため [22], 製品開発には部署や会社といった非常に多くのステークホルダが関わる. そのため, 異なるステークホルダ間で開発ゴールを共有する手段としても SRS が極めて重要な役割を果たすことになる. 一方, 自動車ソフトウェア開発でインスペクションの適用を妨げる 2 つの要因がある.

1 つは自動車ソフトウェアのインスペクションにとって

基準となる参照 SRS 構造と品質特性が確立していないこと, そしてもう 1 つは自動車ソフトウェアの多様性である. Pretschner らは製品ドメインや関連する非機能要求に着目し, 自動車ソフトウェアをインフォテインメント系, ボディ/コンフォート系, セーフティ系, パワートレイン系, インフラ系の 5 つに分類している [22]. 自動車ソフトウェア開発では, これらのドメインごとに異なる知識と技能が求められるため, 異なる組織で担務する開発形態が一般的である. 第一著者の組織では前述の 5 つのドメインとは異なるが, パワートレイン事業部, 電子事業部, 熱事業部, 情報安全事業部に分かれて製品開発を行っている [6]. 異なる開発組織に属するプロジェクトでは独自に SRS の構成要素や品質評価の基準を定義しているため, 開発組織を越えて全社的に統一したインスペクションの仕組みが構築されていない. そのため, インスペクションの成果はインスペクタの経験やドメイン知識に依存してしまう. しかし, 十分な経験と知識を有するドメイン専門家の数は限られてお

^{†1} (株)デンソー
DENSO CORPORATION.

^{†2} 南山大学
Nanzan University

り、すべてのインスペクションに参加できるわけではない。

1.1 本研究の目的

インスペクション活動の効果を全社的に均質化するためには、製品ドメインによらず、すべてのソフトウェア開発組織で体系化されたインスペクション方法を採用すべきである。

体系的なインスペクションに関する先行研究として、ITシステム開発を対象とした第三者インスペクション方法が提案されている [25]。しかし、ITシステムとは異なり多様な製品ドメインをもつ自動車ソフトウェアの開発に、既存の第三者インスペクション方法をそのまま適用することはできない。そのため、本研究の目的は以下の2つである。

- 1) 自動車ソフトウェアの特性に適応した体系的なインスペクション方法を定義する
- 2) 製品ドメインの違いによる SRS の構造的な差異を扱う方法を定義する

以上の目的から、本稿ではドメイン知識を持たない第三者でも製品領域を限定せずに実施可能なインスペクションと、ドメイン知識を持ったメンバ（管理者、開発者、テストなど）から構成されるプロジェクトによるインスペクションとに分けた、二段階インスペクション方法を提案する。

2. 研究の枠組み

2.1 研究のコンテキスト

本研究は自動車部品サプライヤの開発組織において、自動車ソフトウェアシステムを対象に行われたものである。サプライヤは自動車メーカから ECU (Electrical Control Unit) または複数 ECU を組み合わせたシステムの単位で製品開発を受注する。自動車メーカでは複数のサプライヤから納品された ECU やシステムを統合して車両を開発する [15]。

サプライヤは自動車メーカから製品全体に対する要求仕様書を受領し、それをもとにソフトウェア要求仕様書 (SRS)、ハードウェア要求仕様書 (HRS) を作成し製品を開発する。本稿では、そのうちの SRS を研究の対象とする。

SRS の大部分は自然言語によって記述され、必要に応じて UML、データフロー図、決定表などの各種図表で補足される。パワートレインなどの連続制御系では、要求の一部が Simulink などのモデルで表現されることもある。記述言語は主に日本語と英語である。

2.2 研究課題

提案方法の実践と評価を通して以下の研究課題への回答を試みる。

RQ1. 本稿で提案する第三者インスペクションは SRS 品質の改善に有効か？

RQ2. IEEE Std. 830-1998 で推奨されている目次項目は自動車ソフトウェアの SRS にとって適切か？

RQ2.1 自動車ソフトウェアシステムの要求を記述するには不要な目次項目はあるか？

RQ2.2 自動車ソフトウェアシステムの要求を記述するには不足している目次項目はあるか？

2.3 データセット

本研究の対象は第一著者の所属する組織で開発した6つの製品の SRS (SRS_A から SRS_F) と、これから開発を行う製品の SRS (SRS_G) の計7つの仕様書である。各製品の要求仕様書はソフトウェア要求仕様書、ハードウェア要求仕様書など複数のドキュメントから構成される。本稿の対象は、そのうちのソフトウェア要求仕様書である。しかし、ソフトウェアに関する要求仕様書の名称は開発プロジェクトごとに異なっているため、評価者は各ドキュメントの内容と、開発プロセス内でのドキュメントの位置づけから IEEE Std. 830-1998 が定義している SRS に近いドキュメントを選択した。

複数の ECU で構成されているシステム製品の場合は ECU ごとに仕様書が作成されるため、システム製品からは主要な機能を担う ECU を1つ選択し、その ECU の仕様書をインスペクションした。

1つの ECU でも機能ごとに複数の SRS へ分冊される製品もある。これは、自動車ソフトウェアの特徴である仕向けや車両グレードごとの多様なカスタマイズへ対応するためである。SRS が分冊されている場合は、SRS の集まり全体を1つの SRS として評価を行った。表1に評価を行った7つの SRS が対象とする製品のドメインとページ数を示す。すべての SRS は日本語で書かれている (SRS_F のみ英語も併記)。

表1 インスペクション対象 SRS のデータセット

Table 1 Data Set of Inspected SRSs

仕様書 ID	製品ドメイン	ページ数
SRS_A	インフラ	76
SRS_B	ボディ	34
SRS_C	ボディ	39
SRS_D	安全	41
SRS_E	コンフォート	53
SRS_F	安全	108
SRS_G	インフラ	19
合計		370

3. 関連研究

3.1 SRS の構造

SRS に関する国際規格には IEEE Std. 830-1998 [13] がある。その他に、アメリカ国防総省が定めた Software Requirements Specification (DI-IPSC-81433A) [7]、NASA が定めた NASA Software Documentation Standard (NASA-Std-2100-91) [20] などがある。規格以外では、Robertson らがコンサルティング経験から得られた知見をもとに Volere Requirements Specification Template を提案している [23, 24]。Wiegiers らも著書 [28] の中で SRS のテンプレートを提案している。しかし、自動車ソフトウェアの要求に対するこれらのテンプレートの適合性などを評価した研究はない。

3.2 SRS 品質特性と測定方法

SRS 品質特性として、IEEE Std. 830-1998 で定義された 8 つの品質特性が広く用いられている。しかし、IEEE Std. 830-1998 では、各品質特性を測定する方法は述べられていない。Davis らは 24 の SRS 品質特性と各特性を測定するメトリクスを提案している [5]。しかし、これらの測定手法は自然言語で記述された SRS をインスペクションするには具体性に欠け、製品開発にそのまま適用するには不十分である。

3.3 インスペクション方法

Fagan によるインスペクション (Fagan Inspection) [9] の提案以来、その改良に関する多くの提案がある [3]。

3.3.1 N 重インスペクション

N 重インスペクションとは複数の小人数チームでインスペクションを実施する方法である。1 つの大人数チームで実施するよりも多くの欠陥を検出できるという研究がある [18]。しかし、インスペクションの効果はインスペクタの専門知識にもっとも強く依存するという結果も得られている [14]。そのため製品ドメインによらず全社的に統一されたインスペクション方法としては不十分である。

3.3.2 段階的インスペクション

段階的インスペクション [16] は各段階で特定の特性 (移植性, 再利用性, 保守性など) に着目してインスペクションを行う方法である。しかし、ドメイン知識の有無で段階を分けることは想定されていない。また、その適用と効果の検証はソースコードに対してのみ行われており、要求インスペクションへは適用されていない。

3.3.3 インスペクションシステム設計手法

SRS の品質を定量的に測定し、異なるプロジェクト間で SRS 品質の相対比較を行えるインスペクション方法を設計するための方法論 (RISDM) が提案されている [25]。RISDM では標準 SRS およびそこからテラリングされた SRS をインスペクションの対象としている。しかし、自動車ソフトウェア開発では派生開発が主流であり、新製品の SRS も旧製品の SRS をベースに新規点と変化点を反映して作成されるため SRS フォーマットを既存のものから新たなものへと変更するのは容易ではない。そのため、製品ドメインごとに形式の異なる多様な SRS をインスペクションするには RISDM 手法は十分ではない [26]。

3.4 自動車 SRS に関する研究

2001 年の要求工学国際会議 (RE2010) で Weber らが発表した Daimler の自動車開発における要求工学に関する論文 [27] を契機に、自動車分野でも要求工学の重要性が認識されるようになった。その後、欧州の自動車メーカー、主要サプライヤ、および大学を中心に自動車要求工学に関する研究が活発に行われてきた。自動車要求工学に関する研究のうち、SRS を対象とした研究には以下のものがある。

Ott らは Daimler での乗用車向けボディ製品の SRS と関連仕様書の問題 5,999 件を分析し、4 階層の木構造からな

る品質特性モデルを提案している [21]。

Langenfeld らは Bosch での約 5 年間にわたるハイブリッド車向け DC-DC コンバータ SRS の 588 個の欠陥について、IEEE Std. 830-1998 の品質特性と独自に定義した欠陥の発生源に基づき分析している。その結果、品質特性のうち正当性と完全性が要求欠陥の 61% を占め、それらに一貫性を加えた 3 つが最も修正コストが高いことが明らかになった [17]。

Accituna は組込みシステムの SRS 作成者が考慮すべき実行時の状況 (Operational Context) など 6 つの関心事 (Concern) を定義し、各関心事に対するチェックリストを提案している [1]。

しかし、これらの研究では個々の機能要求、非機能要求の記述の良し悪しが対象であり、本研究のように SRS 全体を対象としてドキュメント品質を定量評価する研究は行われていない。

4. アプローチ

1.1 でも述べたとおり、インスペクション方法は製品ドメインによらず、すべてのソフトウェア開発部署で統一されていることが望ましい。しかし、自動車ソフトウェア開発では、製品ドメインごとに異なるドメイン知識が必要となるため、インスペクション方法も製品ドメインの特性を反映する必要がある。この特性に着目し、本稿ではドメイン知識を持たない第三者でも製品領域を限定せずに実施可能なインスペクションと、ドメイン知識を持ったメンバ (管理者, 開発者, テスタなど) から構成されるプロジェクトによるインスペクションとに分けた、二段階インスペクションシステムのアプローチをとる。第三者インスペクションとプロジェクトインスペクションの目的は、それぞれドキュメント品質の確保と要求品質の確保である。これは、要求品質がドキュメント品質に依存し、要求品質を高めるためにはドキュメント品質の確保が前提となるためである。本稿ではこの二段階のうち、第三者インスペクションを主な研究対象とする。

5. 二段階インスペクションシステムのフレームワーク

5.1 二段階インスペクションシステムのプロセス

図 1 に二段階インスペクションシステムのプロセスを示す。第三者インスペクションの目的はドキュメント品質、すなわち記述品質を確保することにある。それに対して、プロジェクトインスペクションは各開発プロジェクトで実施され、要求品質、すなわち内容品質を確保する。

二段階インスペクションシステムでは以下のプロセスに沿ってインスペクションを行う。

- (1) インスペクタによる SRS の受領
要求アナリストや開発チームからなるプロジェクト

から独立したインスペクタが、要求アナリストが作成した SRS を受領する。

- (2) 第三者によるインスペクションの実施
 インスペクタが第三者の立場で SRS のインスペクションを実施する。
- (3) 1次インスペクション結果の判断
 - 3.a) インスペクション結果が合格基準を満たす場合、インスペクタは SRS を開発チームへ引き渡す。
 - 3.b) インスペクション結果が合格基準を満たさない場合、品質スコアと改善アドバイスからなるアセスメントレポートを要求アナリストへ提出する。
- (4) プロジェクトによる SRS の受領
 プロジェクトは、インスペクタが合格基準を満たしていると判断した SRS を受領する。
- (5) プロジェクトによるインスペクションの実施
 プロジェクトを構成する管理者、開発者、テストなどがそれぞれの立場からドメイン知識を活用して SRS のインスペクションを実施する。
- (6) 2次インスペクション結果の判断
 - 6.a) インスペクション結果が合格基準を満たす場合、プロジェクトは SRS をベースライン登録しソフトウェアの開発に着手する。
 - 6.b) インスペクション結果が合格基準を満たさない場合、インスペクションでの指摘をまとめた分析レポートを要求アナリストへ提出する。
- (7) 改善の実施
 - 7.a) アセスメントレポートを受領した場合、要求アナリストは改善アドバイスに従い SRS の更新を行う。
 - 7.b) 分析レポートを受領した場合、要求アナリストは指摘への回答を行い、必要な場合は SRS の改善を実施する。

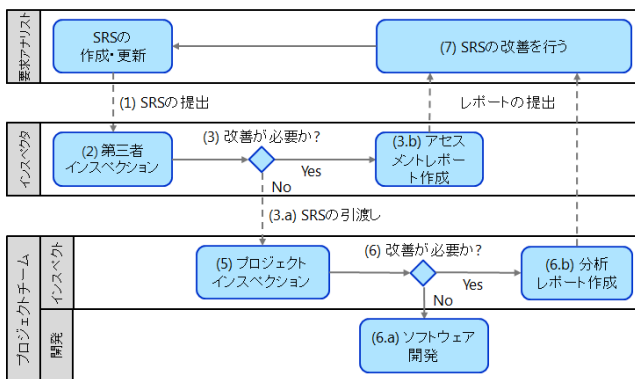


図 1 二段階インスペクションシステムのプロセス

Figure 1 Process of Two-stage Inspection System

5.2 第三者インスペクションのメタモデル

第三者インスペクションのメタモデルを図 2 に示す。

SRS のドキュメント品質を定量的に評価するためには、SRS の構造を定義した参照 SRS とドキュメント品質特性が必要となる。参照 SRS の実体は SRS に含まれるべき要素とその構成を定めた SRS テンプレートである。ドキュメント品質特性は SRS が満たすべき品質特性のうちドメイン知識のない第三者が測定可能な品質特性と定義する。一方、SRS 品質特性からドキュメント品質特性を除いた残りを要求品質特性と定義する。インスペクションマトリクスは参照 SRS の目次項目と測定すべき品質特性を関連付ける。質問セットはドキュメント品質特性を測定するための具体的な手段である。変換マトリクスはプロジェクト SRS の構成要素を参照 SRS へ関連付ける。インスペクションマトリクスと質問セットの組み合わせをインスペクションガイドラインと呼び、ドメイン知識のない第三者がプロジェクト SRS をインスペクトするための指針とする。

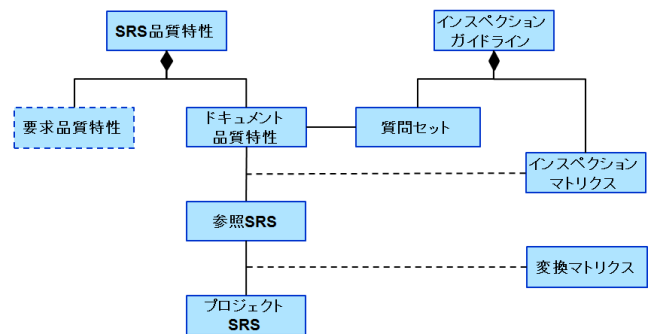


図 2 第三者インスペクションのメタモデル

Figure 2 Third-party Inspection Meta-model

6. 自動車 SRS の第三者インスペクション方法

6.1 第三者インスペクションの目的

第三者インスペクションの目的はドメイン知識を持ったプロジェクトメンバによるインスペクションに先立ち、SRS が十分なドキュメント品質を満たしていることを確認することである。事前に SRS のドキュメント品質を保証することで、プロジェクトメンバは要求品質特性のインスペクションに専念することができる。また、多忙なドメイン専門家もドキュメント品質の不備を指摘する無駄を避けることができるため、要求品質のインスペクションを効率化し、より多くのプロジェクトのインスペクションへ参加できるようになる。

6.2 SRS 品質スコアの測定方法

6.2.1 参照 SRS

自動車ソフトウェアの SRS を記述するための適切な参照 SRS はまだ提案されていない。本稿では、広く使われている SRS テンプレートの 1 つである IEEE Std. 830-1998 の SRS テンプレートを仮の参照 SRS とする。この参照 SRS に基づき、提案方法を実際の自動車ソフトウェア SRS へ適用し、評価することで、IEEE Std. 830-1998 の SRS テンプレ

ートが自動車ソフトウェア SRS のインスペクションのための参照 SRS として適切かを検証する。

6.2.2 SRS 品質特性

IEEE Std. 830-1998 SRS テンプレートを参照 SRS とするため、SRS 品質特性の定義には、同じく IEEE Std. 830-1998 で定義されている品質特性を参考とした。ただし、IEEE Std. 830-1998 で定義されている 8 つの品質特性から重要度/安定度がランク付けされている (Ranked for importance and/or stability) を除外し、あらたに達成可能性 (Achievable) を追加した 8 つの品質特性を SRS 品質特性として定義した。ランク付けを除外したのは、自動車ソフトウェア開発では初期の要求スコープを開発途中で変更することはほとんど行われなためである [10]。逆に、達成可能性を追加した理由は IEEE Std. 830-1998 で定義された品質特性には開発者のパースペクティブが不足しているためである。

6.2.3 ドキュメント品質特性と要求品質特性

二段階インスペクションを実施するためには、測定にドメイン知識を必要とするか否かという観点で SRS 品質特性を 2 つに分類し、どちらかのインスペクションステージへ割り当てる必要がある。測定にドメイン知識を必要としない SRS 品質特性をドキュメント品質特性として第三者インスペクションステージへ割り当てる。一方、測定にドメイン知識を必要とする SRS 品質特性を要求品質特性としてプロジェクトインスペクションステージへ割り当てる。ある SRS 品質特性が単純にどちらかのステージへ割り当てられない場合、その品質特性を分解、または他の SRS 品質特性と統合するなどして新たな品質特性を定義した後、どちらかのインスペクションステージへ割り当てる。各 SRS 品質特性の第三者インスペクションとプロジェクトインスペクションへの割り当てを表 2 に示す。

表 2 SRS 品質特性のインスペクションステージ割り当て

Table 2 Allocation of SRS Quality Characteristics

SRS 品質特性	第三者 インスペクション	プロジェクト インスペクション
正当性	責任追跡性	正当性
無曖昧性	明確性	N/A
検証可能性		
完全性	要求網羅性	記述網羅性
無矛盾性	N/A	無矛盾性
変更容易性	変更容易性	N/A
追跡可能性	追跡可能性	N/A
ランク付け	N/A	N/A
N/A	N/A	達成可能性

6.2.4 インスペクションマトリクス

参照 SRS のすべての要素がすべての品質副特性を満たす必要はない。不要な品質副特性の測定を避けインスペクションを効率化するため、参照 SRS の目次項目と測定すべき品質副特性を関連付けるインスペクションマトリクスを定義する (表 3)。マトリクス上で” X ”が入っている個所

が参照 SRS に対するインスペクションポイントを表している。「1.1 Purpose (目的)」を例にとると、この節に対しては品質副特性 C2-6、C3-3 の評価が必要なことを意味する。インスペクションポイントは合計で 149 か所である。本稿では、各インスペクションポイントで質問への答えがイエス (不備なし) であれば 1 点加点、答えがノー (不備あり) であれば加点しないというスコアリング方法を採用する。

6.2.5 質問セット

ドキュメント品質特性を測定するための具体的な方法としてドメイン知識がなくてもイエスまたはノーで客観的に回答できる質問 (closed question) を定義する。ドキュメント品質特性は抽象度が高くそのままでは質問を定義できないため、抽象度の低い副特性へ分解する。各品質副特性に対応した 15 の質問 (質問セット) を表 4 に示す。

6.2.6 変換マトリクス

プロジェクト SRS 間に存在する構成要素の差異を吸収し統一的な手順でインスペクションを行うため、すべてのプロジェクト SRS の変換先として参照 SRS を定義する。参照 SRS とプロジェクト SRS の対応付けは、プロジェクト SRS の目次項目と参照 SRS の目次項目を対応付ける変換マトリクスを作成することで行う (表 5)。マトリクス上で” X ”が入っている個所が変換ポイントを表している。例えば、プロジェクト SRS の「1.4 I/F とハードウェア前提条件」は参照 SRS の「2.4 Hardware Constraints (ハードウェア

表 3 インスペクションマトリクス

Table 3 Inspection Matrix

IEEE Std. 830-1998 目次項目	品質副特性														
	C1-1	C1-2	C1-3	C1-4	C1-5	C2-1	C2-2	C2-3	C2-4	C2-5	C2-6	C3-1	C3-2	C3-3	C3-4
1. Introduction															
1.1 Purpose											X			X	
1.2 Scope	X													X	X
1.3 Definitions, acronyms, and abbreviations											X			X	
1.4 References														X	
1.5 Overview										X				X	
2. Overall description															
2.1 Product perspective						X					X	X	X	X	X
2.1.1 System interfaces						X					X	X	X	X	X
2.1.2 User interfaces						X					X	X	X	X	X
2.1.3 Hardware interfaces						X					X	X	X	X	X
2.1.4 Software interfaces						X					X	X	X	X	X
2.1.5 Communications intrfaces						X					X	X	X	X	X
2.1.6 Memory constraints						X	X				X	X	X	X	X
2.1.7 Operations						X	X				X	X	X	X	X
2.1.8 Site adaptation requirements						X	X				X	X	X	X	X
2.2 Product functions						X					X	X	X	X	X
2.3 User characteristics						X					X	X	X	X	X
2.4 Constraints						X					X	X	X	X	X
2.5 Assumptions and dependencies						X					X	X	X	X	X
2.6 Apportioning of requirements						X					X	X	X	X	X
3. Specific requirements															
3.1 Extenal interfaces						X					X	X	X	X	X
3.2 Functions															
3.2.1 System Feature															
3.2.1.1 Introduction/Purpose of feature	X	X									X	X	X	X	X
3.2.1.2 Stimulus/Response sequence						X	X	X			X	X	X	X	X
3.2.1.3 Associated functional requirements	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
3.3 Performance requirements	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
3.4 Logical database requirements	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
3.5 Design constraints	X	X				X	X	X	X	X	X	X	X	X	X
3.6 Software system attributes	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
4. Supporting information															X
4.1 Table of contents and index															
4.2 Appendixes															

インスペクションマトリクス

6.3.2 正規化スコア

正規化スコアはプロジェクト SRS の内容を参照 SRS の構造に対応付けた場合のスコアを算出したものである。正規化スコアも非正規化スコアと同様、品質特性ビュー (図 9) と品質副特性ビュー (図 10) および構造ビュー (図 11) から評価できる。正規化スコアの構造ビューからは、参照 SRS の仕様項目を軸にプロジェクト SRS の品質を把握できる。SRS_A では参照 SRS の 1, 2, 4 章に相当する部分の品質スコアが 3 章に比べて相対的に低いことがわかる。

6.3.3 仕様項目ヒストグラム

プロジェクト SRS を評価するには変換マトリクスを作成する。変換マトリクスに出現する“X”の数を参照 SRS の仕様項目ごとに集計したのが仕様項目ヒストグラム (図 3) である。正規化スコアからは参照 SRS のどの仕様項目に相当する部分のスコアが低いかは把握できる。しかし、スコアが低い原因が不十分な記述内容にあるのか、その参照 SRS 仕様項目に相当する仕様項目がプロジェクト SRS には含まれていないのかを判別することはできない。仕様項目ヒストグラムを用いることでその判断が可能となる。

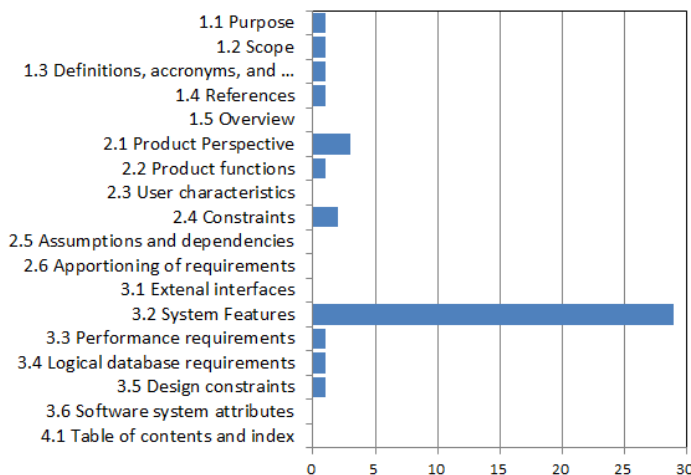


図 3 SRS_A の仕様項目ヒストグラム

Figure 3 Document Histogram for SRS_A

6.3.4 スコアヒートマップ

スコアヒートマップ (図 12) はインスペクションポイント全体での品質スコアの分布を表す。スコアヒートマップからは、正規化スコアの構造ビューと品質特性ビューからは読み取れない、ある仕様項目内での品質特性ごとのスコアの分布と、逆にある品質特性内での仕様項目ごとの分布が把握できる。スコアヒートマップの各セルの数字はそのインスペクションポイントの品質スコアである。インスペクションの対象ではないセルには数字ではなくハイフンが示されている。

7. 実践と評価

7.1 提案手法の効果検証 (RQ1)

7.1.1 検証目的

本稿で提案する第三者インスペクションによって得られ

る SRS 品質スコア情報が要求アナリストによるプロジェクト SRS の改善に対し有効であるかを検証する。

7.1.2 検証方法

表 1 に示したデータセットのうち SRS_A と SRS_G は同一の開発チームにより作成された。このチームでは要求仕様書の改善活動に取り組んでおり、SRS_G は SRS_A に対するインスペクション結果 (6.3 参照) を踏まえて作成された要求仕様書である。ただし、SRS_G は SRS_A の版を改定したものではなく、異なる製品向けに作成された新規の要求仕様書である。これら 2 つ要求仕様書 (SRS_A と SRS_G) の品質スコアを比較する。

7.1.3 検証結果

SRS_A と SRS_G の品質スコアを品質特性ビューと構造ビューで比較した結果をそれぞれ図 13 と図 14 に示す。

SRS_A と SRS_G は異なる構造を持つため、図 13 と図 14 の比較結果はともに正規化スコアによるものである。図 13 の品質特性ビューからは、SRS_G では SRS_A に比べて責任追跡性 (C1) のスコアが向上しているが、明確性 (C2)、記述網羅性 (C3)、追跡可能性 (C5) の 3 特性ではほぼ変化がなく、変更容易性 (C4) では逆にスコアが低下していることがわかる。次に図 14 の構造ビューによる比較を見ると、品質特性の場合と同様にスコアが向上している参照 SRS 仕様項目もあれば低下している項目もある。

続いてスコアヒートマップ (図 15) により SRS_A から SRS_G への品質スコアの増減を大域的に視覚化する。図 15 は各インスペクションポイントに対し SRS_G の品質スコアから SRS_A の品質スコアを減じた結果を表している。このスコアヒートマップからは、SRS 全体で品質スコアが増減した品質特性はなく、品質スコアの変動は主に仕様項目の増減に起因していることが読み取れる。

これらの分析結果から、SRS_G が SRS_A よりも改善されたとは言えない。これは、先行研究[25]でも示されているように、1 回の SRS の改訂では必ずしも十分な改善が達成されないためと考えられる。そのような場合でも、従来は主観のみで行っていた要求文書の良し悪しの判断が品質スコアを用いることで客観的に行えるようになるため、本稿で提案する第三者インスペクションから得られる品質スコアは SRS の改善結果を判断するには有効であるといえる。

7.2 参照 SRS の妥当性検証：不要項目の検証 (RQ2.1)

7.2.1 検証目的

自動車ソフトウェアの要求を記述するうえで IEEE Std. 830-1998 SRS テンプレートの仕様項目には不要な要素がないかを検証する。

7.2.2 検証方法

7 つの SRS それぞれに作成された仕様項目ヒストグラムをもとに作成した箱ひげ図 (図 4) から、SRS 記述量の分布を分析する。タイトル名の右端に * の付いている仕様項目 (3.1, 3.2, 3.3, 3.4, 3.6) は一般的に仕様書のページ数

が増えるほど出現数が多くなる。そのため、それらの項目に関しては、以下の式 (1) で定義される記述密度を示す。

$$\text{記述密度} = \frac{\text{仕様項目の出現数}}{\text{仕様項目のページ数}} \quad (1)$$

7.2.3 検証結果

正規化された項目を除くと、最小値が 1 未満の項目は少なくとも 1 つの SRS がその項目を含んでいないことを意味する。そのなかでも、中央値が 1 未満のものはデータセットすべての SRS で記述量が少ないことを表している。その観点で見ると、すべての SRS で十分な記述が行われている項目はないことがわかる。そこで、1 章と 2 章について十分な記述が行われていない理由について考察する。

1 章の記述が少ないのは、ECU 単品で受注した製品の場合には、システム全体の目的といった 1 章に書かれるべき情報はあまり重要性ではないと多くのプロジェクトでは認識している可能性がある。

2 章については、すべての仕様書でまったく記述のなかった項目について考察する。これらの項目に関する記述がない理由は自動車ソフトウェアシステムの開発形態に起因する可能性がある。

まず、「2.3 User characteristics (ユーザ特性)」について考察する。自動車ソフトウェアではユーザインタフェースを持つものはナビゲーションソフトウェアなどに限定されるため、この項目の出現頻度が低くなったと考えられる。今回の評価サンプルの中にはユーザインタフェースを持つ製品がなかった。

次に「2.5 Assumptions and dependencies (前提と依存関係)」と「2.6 Apportioning of requirements (要求の割り当て)」について考察する。自動車ソフトウェアの開発では、試作フェーズを繰り返した後に最終的な量産ソフトウェアを完成

させる。試作フェーズごとに自動車メーカーから新たな要求が変更要求仕様書として提供される [15]。このように試作フェーズごとの要求の追加と変更が前提で開発が行われていることが、これらの項目が記述されない要因とも考えられる。また、自動車ソフトウェアの開発では、現時点では IT システムほど要求の変動が大きくないことも要因として考えられる。

SRS 記述量の分布分析から、多くの自動車ソフトウェア開発において「ユーザ特性」、「前提と依存関係」、「要求の割り当て」は必須の要素ではない可能性があることが分かった。これらの要素は、自動車ソフトウェアの要求をより適切に扱うため、類似しているが異なる観点へ置き換える必要性を示唆している。例えば、自動車ソフトウェアのインタフェース対象は人間のユーザとは限らないため、インタフェースの特性をより一般化するよう、ユーザ特性は環境特性などと置き換えることが望ましい。

7.3 参照 SRS の妥当性検証：不足項目の検証 (RQ2.2)

7.3.1 検証目的

自動車ソフトウェアの SRS を記述するうえで IEEE Std. 830-1998 SRS テンプレートの仕様項目には不足している要素がないかを検証する。

7.3.2 検証方法

IEEE Std. 830-1998 を参照 SRS として 7 つのプロジェクト SRS に対する変換マトリクスを作成した。その際に、参照 SRS の仕様項目に適切な対応先が見つからなかった要素についてその原因を分析する。

7.3.3 検証結果

自動車ソフトウェアの特徴のうち IEEE Std. 830-1998 では適切に扱うのが難しい 2 つの特徴を特定した。

(1) 多様なバリエーション要求

インスペクションした 7 つの SRS のうち新規開発品向けの SRS_G を除く 6 つの SRS には特定の車種や仕向けのみで有効な要求が含まれていた。これは、自動車ソフトウェア開発には車種ごとのグレード間での差別化などのために非常に多くのバリエーションが要求されるためである。典型的なプレミアム車種では約 80 のバリエーションを持つ [22]。このように多様なバリエーションを扱う必要があるため、近年の自動車ソフトウェア開発ではプロダクトライン開発が取り入れられつつある。しかし IEEE Std. 830-1998 ではソフトウェアプロダクトライン開発にとって重要な共通性 (commonality) と可変性 (variability) の扱いが考慮されていない。

(2) 抽象度の異なる要求

製品の機能要求はすべて参照 SRS の「3.2 System Features (機能)」に属する。しかし、本研究でインスペクションした SRS には、システム全体の振る舞いを表した抽象的な要求から、設計の領域まで踏み込んだ詳細な要求まで様々な抽象度の要求が存在した。一般的な自動車ソフトウェアは

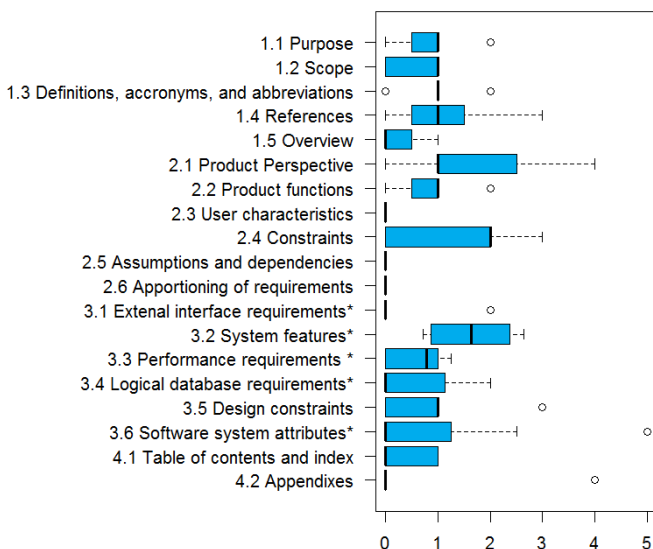


図 4 SRS 記述量の分布

Figure 4 Distribution in Amount of SRS Description

図 5 に示すような階層アーキテクチャを持ち、COTS (Commercial Off-The-Shelf) 製品を利用する場合を除くと、すべてのレイヤが開発対象となる。そのため、抽象度の高いアプリケーションレイヤに関しては SRS が設計情報を含まないようにする必要がある [5, 13]。一方、抽象度の低いドライバレイヤでは SRS の記述も設計まで踏み込む必要がある [27]。したがって、自動車ソフトウェアの SRS では高レベルの要求から低レベルの要求まで抽象度の異なる要求を単一製品の SRS の中で扱う必要がある。しかし、IEEE Std. 830-1998 の目次項目と記述ガイドラインは抽象度の異なる要求を適切に構造化するには不十分である。

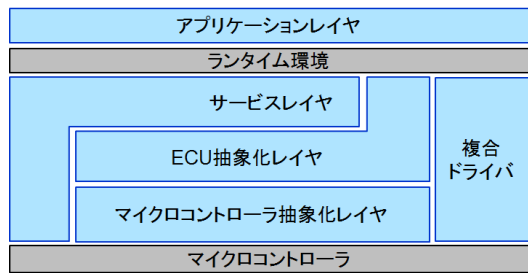


図 5 AUTOSAR 階層ソフトウェアアーキテクチャ [2]
Figure 5 AUTOSAR Layered Software Architecture [2]

8. 考察

8.1 二段階インスペクションの適用性

IT システムの SRS インスペクションを対象とした先行研究 [25]では、ドメイン知識を有さない第三者でもすべての品質特性を測定している。しかし、自動車ソフトウェアは、それ自体に多様なドメインを内包し、インスペクションにはそれらのドメインごとに固有の知識が必要となる。そのため、IT システムの SRS インスペクション方法をそのまま適用することは困難である。そこで、本研究ではインスペクションをドメイン知識が必要な領域と不要な領域に分け、順次実施する二段階インスペクションシステムを提案した。さらに、1 次インスペクションステージである第三者インスペクションの具体的な方法を定義し、実際の仕様書に対して提案方法を実践し有効性を評価した。実践結果から、異なる製品ドメインの SRS に対して第三者が統一的手段でインスペクションを実施できること、品質スコアは SRS の改善結果を判断するには有効であることが確認された。ドメイン知識がインスペクションに必要なことは自動車ソフトウェアに限らず、組み込みソフトウェア全般に言えることである。そのため、本稿で提案した二段階インスペクションは多くの組み込みソフトウェア開発へ適用可能な方法であると考えられる。

8.2 自動車ソフトウェア要求の特徴

実際の 7 つの自動車製品ソフトウェアの SRS 記述量の分を分析し、IEEE Std. 830-1998 の SRS テンプレート項目の中で自動車ソフトウェアの要求には適合しない可能性のあ

る項目を特定した。一方、自動車ソフトウェアの要求にとっては重要であるが IEEE Std. 830-1998 の SRS テンプレートと記述ガイドラインでは適切に扱うことのできない項目として次の 2 つの特徴を特定した。

- 1) 多様なバリエーション要求
- 2) 抽象度の異なる要求

自動車ソフトウェアの要求に最適な SRS の構造に関する提案はない。本研究により得られた知見は今後のこの分野での研究に対する 1 つの指針となりえる。

9. まとめ

本稿では第三者インスペクションとプロジェクトインスペクションから構成される二段階インスペクションシステムを提案した。この中で、第三者インスペクションの具体的な方法を提案し、その効果を評価するため 7 つの SRS に対して提案方法を適用した。

第三者インスペクションから得られた品質スコア情報は SRS 品質に対して客観的な視点を与えるため、SRS の改善結果を判断するために有効であることが確認できた。7 つの SRS に対する記述密度の分析からは、「ユーザ特性」、「前提と依存関係」、「要求の割り当て」の項目は自動車ソフトウェアの SRS には適切ではない可能性が明らかになった。また、IEEE Std. 830-1998 SRS テンプレートは自動車ソフトウェアの特徴である多様なバリエーション要求、抽象度の異なる要求を適切に構造化して扱うためには不十分であるという知見も得た。

本稿では、二段階インスペクションシステムのうち第三者インスペクション部分のみを定義することにどまったが、プロジェクトインスペクションについても実用的な方法を検討し、二段階インスペクションシステム全体の有効性の検証にも取り組む予定である。

謝辞 (一社) 情報サービス産業協会 (JISA) 要求インスペクション設計研究会の斎藤忍委員はじめ、ご討議頂いた委員各位に感謝します。

参考文献

- [1] D. Aceituna, Survey of Concerns in Embedded Systems Requirements Engineering, SAE Int'l J. of Passenger Cars- Electronic and Electrical Systems, Vol. 7, No. 1, May 2014, pp. 1-7.
- [2] AUTOSAR Consortium, AUTOSAR – Layered Software Architecture. AUTOSAR Consortium, Tech. Rep. Release 4.2.2, Jul. 2015.
- [3] A. Aurum, H. Petersson, and C. Wohlin, State-of-the-Art: Software Inspections after 25 Years, J. of Software Testing Verification and Reliability, Vol. 12, No. 3, Sep. 2002, pp. 133-154.
- [4] P. Braun, et al., Guiding Requirements Engineering for Software-Intensive Embedded Systems in the Automotive Industry, Computer Science - Research and Development, Vol. 29, No. 1, Springer, Feb. 2014, pp. 21-43.
- [5] A. Davis, et al., Identifying and Measuring Quality in a Software Requirements Specification, Proc. of the First Int'l Software Metrics Symposium, IEEE Computer Society, May 1993, pp. 141-152.

[6] DENSO CORPORATION., <http://www.denso.co.jp/ja/>.

[7] DoD, DI-IPSC-81433A, Data Item Description: Software Requirements Specification, Dec. 1999.

[8] C. Ebert and C. Jones, Embedded Software: Facts, Figures, and Future, IEEE Computer, Vol. 42, No. 4, Apr. 2009, pp. 42-52.

[9] M. Fagan, Design and Code Inspections to Reduce Errors in Program Development, IBM Systems Journal, Vol. 15. No. 3, 1976, pp. 182-211.

[10] G. Fanmuy, et al., Requirements Verification in the Industry, Proc. of CSDM 2011, Springer, Dec. 2011, pp. 145-160.

[11] G. Fanmuy and G. Foughali, A Survey on Industrial Practices in Requirements Engineering, INCOSE International Symposium, Vol. 22, No. 1, Jul. 2012, pp. 1021-1040.

[12] K. Grimm, Software Technology in an Automotive Company - Major Challenges, Proc. of ICSE 2003, IEEE Computer Society, May 2003, pp. 498-503.

[13] IEEE, Std. 830-1998: IEEE Recommended Practice for Software Requirements Specifications, IEEE Computer Society, 1998.

[14] E. Kantorowitz, A. Guttman, and L. Arzi, The Performance of the N-Fold Requirement Inspection Method, J. of Requirements Engineering, Vol. 2, No. 3, 1997, pp. 152-164.

[15] J. Krisch, and F. Houdek, The Myth of Bad Passive Voice and Weak Words: An Empirical Investigation in the Automotive Industry. Proc. of RE 2015, IEEE Computer Society, Aug. 2015, pp. 344-351.

[16] J. C. Knight, and E. A. Myers, An Improved Inspection Technique, CACM, Vol. 36, No. 11, Nov. 1993, pp. 51-61.

[17] V. Langenfeld, et al., Requirements Defects over a Project Lifetime: An Empirical Analysis of Defect Data from 5-Year Automotive Project at Bosch, Proc. of REFSQ 2016, LNCS Vol. 9619, Springer, Mar. 2016, pp. 145-160.

[18] J. Martin, and W. T. Tsai, N-Fold Inspection: A Requirements Analysis Technique, CACM, Vol. 33, No. 2, Feb. 1990, pp. 225-232.

[19] S. McConnell, The Best Influences on Software Engineering, IEEE Software, Vol. 17, No. 1, Jan./Feb. 2000, pp. 10-17.

[20] NASA, NASA-STD-2100-91, NASA Software Documentation Standard, NASA Headquarters Software Engineering Program, Jul. 1991.

[21] D. Ott, Defects in Natural Languages Requirements Specifications at Mercedes Benz, Proc. of RE 2012, IEEE, Sep. 2012, pp. 291-296.

[22] A. Pretschner, et al., Software Engineering for Automotive Systems: A Roadmap, Proc. of FOSE 2007 (Future of Software Engineering), IEEE Computer Society, May 2007, pp. 55-71.

[23] J. Robertson, and S. Robertson, Volere Requirements Specification Template Edition 17, Atlantic Systems Guild, 2014.

[24] S. Robertson, and J. Robertson, Mastering the Requirements Process, 2nd ed., Addison-Wesley, 2006.

[25] S. Saito, M. Takeuchi, S. Yamada, and M. Aoyama, RISDM: A Requirements Inspection Systems Design Methodology, Proc. of RE 2014, IEEE Computer Society, Aug. 2014, pp. 223-232.

[26] A. Takoshima, and M. Aoyama, Assessing the Quality of Software Requirements Specifications for Automotive Software Systems, Proc. of APSEC 2015, IEEE CPS, Dec. 2015, pp. 393-400.

[27] M. Weber and J. Weisbrod, Requirements Engineering in Automotive Development: Experiences and Challenges, IEEE Software, Vol. 20, No. 1, Jan./Feb. 2003, pp. 16-24.

[28] K. Wiegers and J. Beatty, Software Requirements, 3rd ed., Microsoft Press, 2013.

表 6 SRS_D の変換マトリクス
 Table 6 Translation Matrix for SRS_D

	1.1 Purpose	1.2 Scope	1.3 Definitions, acronyms, and abbreviations	1.4 References	1.5 Overview	2.1 Product Perspective	2.1.1 System Interfaces	2.1.2 User Interfaces	2.1.3 Hardware Interfaces	2.1.4 Software Interfaces	2.1.5 Communications Interfaces	2.1.6 Memory constraints	2.1.7 Operations	2.1.8 Site adaptation requirements	2.2 Product functions	2.3 User characteristics	2.4 Constraints	2.5 Assumptions and dependencies	3.1 External interfaces	3.2 Apportioning of requirements	3.3 Performance requirements	3.3.1 Introduction/Purpose of feature	3.3.2 Stimulus/Response sequences	3.3.3 Associated functional requirements	3.3.4 Logical database requirements	3.6 Software system attributes	4.2 Appendices	
																											X	
1	1.1																											
	1.2																											
		1.2.1																										
		1.2.2																										
		1.2.3																										
	1.3																											
		1.3.1																										
		1.3.2																										
		1.3.3																										
	1.4																											
	1.5																											
2	2.1																											
		2.1.1																										
		2.1.2																										
		2.1.3																										
	2.2																											
	2.3																											
3	3.1																											
		3.1.1																										
		3.1.2																										
		3.1.3																										
		3.1.4																										
		3.1.5																										
	3.2																											
		3.2.1																										
		3.2.2																										
		3.2.3																										
		3.2.4																										
	3.3																											
4	4.1.1																											
		4.1.1.1																										
		4.1.1.2																										
		4.1.1.3																										
		4.1.1.4																										
	4.1.2																											
	4.1.3																											
	4.1.4																											
	4.1.5																											
	4.1.6																											
	4.1.7																											
	4.1.8																											
5	5.1.1																											
	5.1.2																											
	5.1.3																											
	5.1.4																											
	5.1.5																											
6	6.1.1																											
	6.1.2																											
7	7.1.1																											
	7.1.2																											

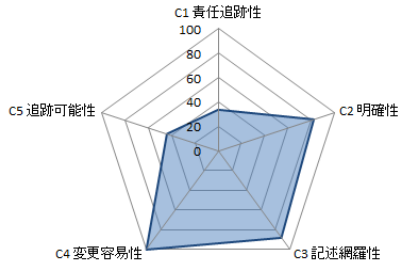


図 6 SRS_A の非正規化スコア (品質特性ビュー)

Figure 6 Non-normalized Quality Characteristics View for SRS_A

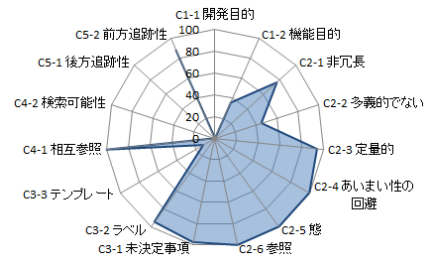


図 7 SRS_A の非正規化スコア (品質副特性ビュー)

Figure 7 Non-normalized Quality Sub-Characteristics View for SRS_A

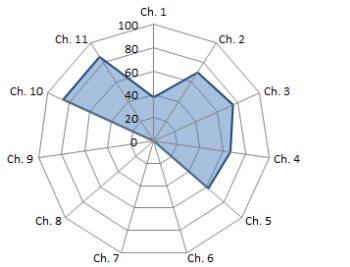


図 8 SRS_A の非正規化スコア (構造ビュー)

Figure 8 Non-normalized Structural View for SRS_A

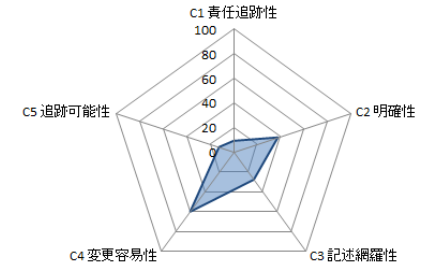


図 9 SRS_A の正規化スコア (品質特性ビュー)

Figure 9 Normalized Quality Characteristics View for SRS_A

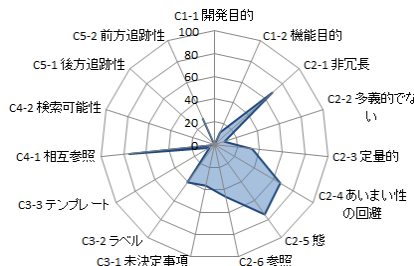


図 10 SRS_A の正規化スコア (品質副特性ビュー)

Figure 10 Normalized Quality Sub-Characteristics View for SRS_A

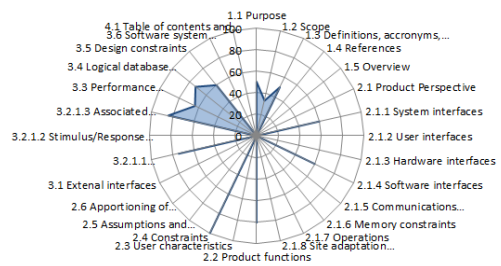


図 11 SRS_A の正規化スコア (構造ビュー)

Figure 11 Normalized Structural View for SRS_A

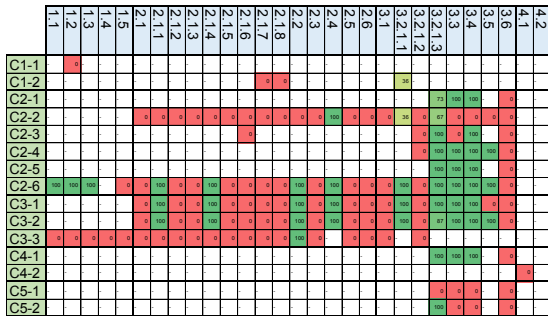


図 12 SRS_A のスコアヒートマップ

Figure 12 Score Heat Map for SRS_A

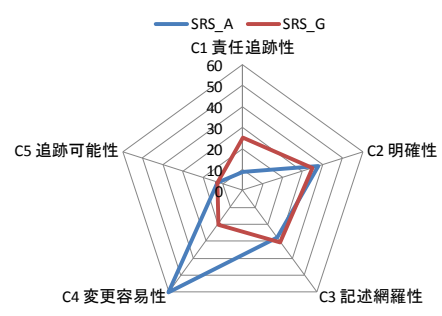


図 13 SRS 品質の比較 (品質特性ビュー)

Figure 13 Relative Comparison in Quality Characteristics View

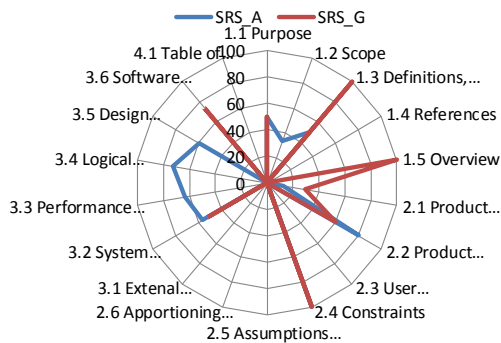


図 14 SRS 品質の比較 (構造ビュー)

Figure 14 Relative Comparison in Structural View

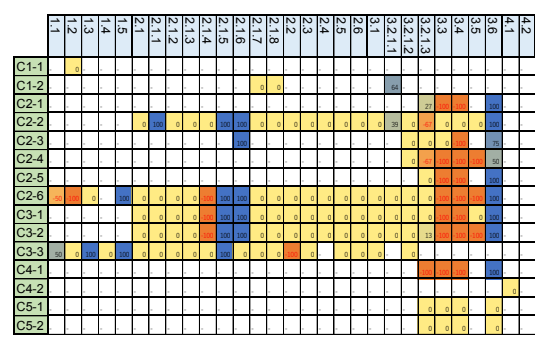


図 15 SRS_A 品質の比較 (スコアヒートマップ)

Figure 15 Relative Comparison in Score Heat Map