

# 企業グループ横断的な組み込みソフトウェア設計力の 評価および強化施策の実践

小川 秀人<sup>1,a)</sup> 島袋 潤<sup>2,b)</sup> 日野 雅透<sup>3,c)</sup> 若林 昇<sup>1,d)</sup> 川上 真澄<sup>1,e)</sup>

受付日 2022年4月21日, 採録日 2024年4月15日

**概要:** ネットワーク接続やデータ活用などの機能の拡大により、いわゆる組み込みソフトウェアの規模が増大し、ソフトウェアの質も変化していると言われて久しい。このようなソフトウェアの量と質の変化に対応するためには、ソフトウェア開発組織の有するソフトウェア設計能力の強化が求められる。このとき、ソフトウェア開発組織はソフトウェアの設計・実装といったソフトウェア開発そのものの能力だけでなく、組織マネジメントなども含めた幅広い能力が必要となる。組織能力を強化するためには、現状の設計力を評価し、達成目標を定め、目標を達成するための強化策を策定・実践し、強化策の効果を評価し、さらに次の強化を進める PDCA サイクルの実践が重要である。日立グループでは、このようなソフトウェア設計力強化を組織横断的に実践する取り組みを進めてきた。本稿では、企業グループ横断的な組み込みソフトウェア設計力強化に向けた現状把握と強化目標設定のための評価手法と、それに基づく設計力強化の実践について述べる。

**キーワード:** 組み込みソフトウェア, ソフトウェアプロセス, プロセスのモニタリングと制御, プロセス改善

## A Practice of Evaluating and Strengthening Capabilities in Developing Embedded Software Across A Corporate Group

HIDETO OGAWA<sup>1,a)</sup> JUN SHIMABUKURO<sup>2,b)</sup> MASAYUKI HINO<sup>3,c)</sup> NOBORU WAKABAYASHI<sup>1,d)</sup> MASUMI KAWAKAMI<sup>1,e)</sup>

Received: April 21, 2022, Accepted: April 15, 2024

**Abstract:** The scale of the so-called embedded software increases, and the characteristics of software are changing by the expansion of the new features such as network connection and data utilization. In order to respond to such changes in the quantity and quality of software, it is necessary to strengthen the development capabilities of the software development organizations. Software development organizations require a wide range of capabilities, including the development capabilities of software design and implementation, as well as organizational management. In order to strengthen the development capabilities of the organization, it is important to evaluate the current development capabilities, set goals for achievement, formulate and practice reinforcement measures to achieve targets, evaluate the effectiveness of the reinforcement measures, and practice the PDCA cycle that further strengthens the following reinforcement. The Hitachi Group has been implementing efforts to strengthen the softwares development capabilities across the business organizations. In this paper, we describe the evaluation and strengthening method of the development ability based on the current evaluation and target settings.

**Keywords:** embedded software, software process, process monitoring and control, process improvement

<sup>1</sup> 株式会社日立製作所 研究開発グループ  
Hitachi,Ltd., Yokohama, Kanagawa 244-0817, Japan  
<sup>2</sup> 株式会社日立製作所 マネージド&プラットフォームサービス事業部  
Hitachi,Ltd., Yokohama, Kanagawa 244-0817, Japan  
<sup>3</sup> 株式会社日立製作所 サービス・ソフト品質強化本部  
Hitachi,Ltd., Chiyoda, Tokyo 110-0006, Japan  
a) hideto.ogawa.cp@hitachi.com  
b) jun.shimabukuro.qw@hitachi.com  
c) masayuki.hino.dx@hitachi.com  
d) noboru.wakabayashi.mu@hitachi.com  
e) masumi.kawakami.ch@hitachi.com

### 1. はじめに

日本国内における組み込みソフトウェア開発に対する問題意識はたびたび指摘されている。2000年代には旧来の機械あるいはアナログ回路による機器の制御がソフトウェアによるデジタル制御に変わり、組み込みソフトウェアの開発が重要となり、技術者不足や品質確保が大きな問題と

なった [1]。当時はリアルタイム性や省リソースなどの組み込みソフトウェアの技術特性に加えて、ソフトウェア開発の経験がない事業主体においてソフトウェア開発が必要となり、ソフトウェア開発組織の立ち上げやソフトウェア開発技術の定着も課題となった。日立グループにおいても、組み込みソフトウェア設計力強化のための全社的な取り組みを推進してきた [2]。

2010年頃からは組み込みソフトウェアの特性がさらに変化し多様化している。組み込みソフトウェアが利用される製品分野が広がり、ビジネスモデルやステークホルダも多様化し、求められる要求や開発における関心事も製品により大きく異なるようになった [3]。近年は機器を正しく制御するだけでなく、ユーザが快適に使える官能性や、少ないエネルギーで制御できる環境性など、非機能性能を向上することがソフトウェアの要件となってきている。そのために、種々のセンサから得られる複数の情報を統合するセンサフュージョンによる制御や、機械学習に基づく制御など、制御方式も多様化している。さらには、組み込み機器から得られたデータをネットワーク経由で収集する IoT (Internet of Things) の広がりや、収集されたデータに基づく遠隔管理などの IT サービスの実現、データ解析に基づく機器運用の最適化など、組み込みソフトウェアには機器に閉じない制御以外の機能性も求められている。このように、組み込みソフトウェアに取り巻く環境が大きく変化し続けていることから、組み込みソフトウェア開発組織がもつ開発能力は継続的に強化していかなければならず、日立グループにおける全社的な改善の取り組みも継続的に推進している [4]。

ソフトウェア開発能力の向上のためには、まず組織の能力を評価することが重要である。そのためには組織能力を評価する客観的な評価基準が必要である。また前出のようなソフトウェア開発に求められる環境の変化に伴い、ソフトウェア設計力の評価観点の継続的な見直しも必要となる。組織の設計力を評価したうえで、組織としての強化目標を立てるが、組織のビジネスモデルや開発する製品の特性により、強化すべき項目の優先順位が異なるため、組織ごとの目標設定が重要である。目標を設定することでその目標を達成するための具体的な施策を検討することができる。さらには施策を実行したのちソフトウェア設計力を再評価することで目標の達成度合いを明らかにし、新たな施策の検討や目標の立案により、段階的に組織能力を向上できると考えられる。

日立グループのように複数の事業体からなりそれぞれが多様な製品を開発している場合、事業や製品によって異なる特性をもつため、一律の評価指標や施策を定めることは困難である一方、複数の事業や製品の評価を相対的に比較することでグループとしての弱点を把握することや、ある事業や製品で有効であった施策を他の事業や製品に適用し

改善を図ることが設計力強化の有効な手段となりうる。

本稿では、企業グループにおける組み込みソフトウェア設計力強化施策の1つとして実践している、設計力評価と目標設定の取り組みについて報告する。以降、2章では本取り組みの背景を示し、3章で考え方を示す。4章ではソフトウェア設計力評価手法としてのアセスメントシートについて提案し、5章で評価・強化を推進する組織体制について述べる。また、セルフアセスメントを実践するための Web 環境について6章に示す。これらの提案手法を実践した結果を7章で示したうえで、8章において評価結果の組織横断的な解釈について論じ、9章にて本稿をまとめる。

## 2. 背景

ソフトウェア開発プロセスを評価する枠組みの1つとして、SPICEとして知られる ISO/IEC 15504 [5] やその後継規格である ISO/IEC 33020 [6] などがある。また特定の産業分野向けに特化した枠組みの一例として、Automotive SPICE [7] が自動車業界では用いられている。これらの規格に基づくアセスメントによるソフトウェア開発組織の成熟度診断は、調達条件の一部となるなどソフトウェア事業において重要な役割を果たしている。事業に必要な特定の成熟度レベルを獲得するための活動が、組織のソフトウェア設計力の向上に役立っていると言える。しかし、これらの正規のアセスメントには人的・時間的リソースを多く必要とし、継続的なソフトウェア開発力の強化活動に用いることは難しい側面がある。また、事業に必要な特定のレベルに到達することが目的化し、継続的な設計力強化に結びつかないこともある。それらの規格は組織の成熟度を主にプロセス的なアクティビティの有無で評価するため、ソフトウェアの設計や開発のための具体的な技法が成熟度とどう関係しているかが必ずしも明確ではない。また事業におけるソフトウェア開発では、これらの成熟度評価のほか、機能安全規格 [8] やセキュリティ規格 [9] など準拠すべき規格は多々あり、それらの整合性を取りながらも効率的な開発環境を確立する必要がある。さらには、日立グループのように多様な事業分野で多様な製品開発を行っている企業では、個々の製品の特性に配慮しながらも組織横断的な強化活動を行うことが求められる。

これらの背景を踏まえて、筆者らは企業グループ内の複数の組織を対象とした組み込みソフトウェア設計力の強化を図るため、ソフトウェア設計のための技術や仕組みに着目した設計力評価方式を策定し、組織横断的な取り組みとしてソフトウェア設計力の評価と強化を実践してきた [10]。なお、ここでいう設計力とは、ソフトウェアの仕様や実装を策定する能力だけでなく、それらを含みソフトウェア開発組織が有すべき技術的な能力全般を意味している。

### 3. 組織横断的な設計力強化の考え方

組織改善の方法には問題解決型と目標達成型の2つの考え方があると言われる。問題解決型とは、組織活動において発現した問題の原因を追求し、その原因を解決することにより、問題の再発を防止する方法である。問題解決型の組織改善は類似問題の解決には効果的である反面、場あたりの対策となる可能性がある。目標達成型とは、改善目標を定め目標を達成するための施策を実践する方法である。目標達成型の組織改善は取り組みの方向性を共有し組織的な施策を実践できる反面、目標が現実と乖離すると活動が形骸化する可能性がある。また、多数の組織を跨る組織横断的な改善活動においては、すべてのサブ組織に対して一律に設定した目標の下で活動を行うことは非現実的である。

日立グループでは、全社を統括する品質保証部門に製品やサービスの品質情報が集約されており、これには製品開発中の品質状況や製品出荷後の品質状況も含まれる。これらの情報に基づき、品質面において特に強化を要する組織に対して直接的な課題解決を行っている。これは問題解決型改善活動である。一方で各組織の抱える課題を抽出し継続的に改善するため、本稿に示すソフトウェア設計力の評価を行う。また評価結果に対して改善目標を立て、改善施策を実施し、定期的な評価を行う。これは目標達成型の改善であると言える。さらに、各組織での評価結果と改善目標と改善のために実施した取り組みを全社組織に集約する。これにより、企業グループ全体で共通に抱える課題が表出化されるとともに、集積される対策は問題解決型改善にも利用される。

このように問題解決型と目標達成型の改善を同時に行うことで、個別事象の解決と、組織単位の改善と、企業グループ横断的なソフトウェア設計能力の向上を図っている。本稿で示す設計力評価とそれに基づく改善活動は、目標達成型改善を行いつつ問題解決型改善を支援する役割を果たすものである。

### 4. ソフトウェア設計力評価手法

#### 4.1 設計力評価手法の概要

組織の設計力を向上するためには、当該組織が有している能力と抱えている課題を把握する必要がある。そのため、組み込みソフトウェア開発を対象として想定したアセスメント手法を策定した。

本アセスメント手法では「設計力評価シート」と呼ぶアセスメントシートを用いる。設計力評価シートは、次の4つの軸に分類される評価項目と、それぞれの評価項目に対する5段階の技術レベルからなる。評価軸は以下の4つの軸に分類される：

P 開発プロセス

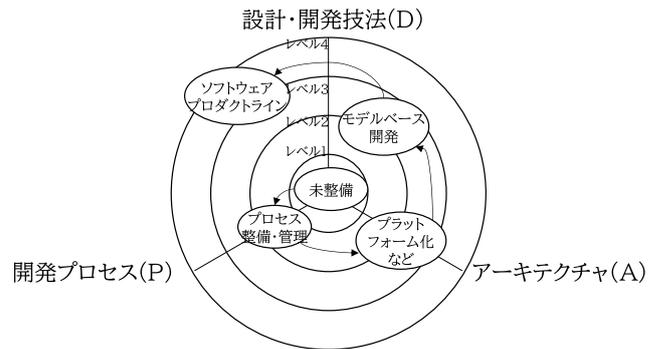


図1 組織改善のスパイラルモデル [2]

Fig. 1 Spiral model of organizational improvement.

- A アーキテクチャ
- D 設計・開発方式
- S 安心・安全

設計力強化が開発プロセスの整備 (P)、アーキテクチャの整備 (A)、設計・開発方式 (D) の順にステップアップし、これらを繰り返すことで成長するスパイラルモデル (図1) を仮説していることから、P,A,Dの軸を設けている。これに、組み込みソフトウェアで重要となる安全性・信頼性に関わるSを加えた4軸としている。

5段階の技術レベルは、各評価軸に対して実践されている技術レベルを示す：

- Level0 組織的な施策がとられていない状態
- Level1 最低限の施策がとられているレベル。意図的に留まる場合を除いて改善が必要
- Level2 改革活動・強化活動において、最低限期待するレベル
- Level3 当該企業グループ内でのベストプラクティスと見込まれるレベル
- Level4 従来型の「決まったものを確実に作る」開発において、実用段階で最も高いレベル

これらの評価軸と技術レベルに対して、利用・実践することが求められる技術や施策を定義して、それらを実践しているかどうかを評価することで、組織の設計力を評価する。表1にアセスメントシートの構成の概要を示す。縦軸に4つの軸に分類された評価項目を、横軸には技術レベルをそれぞれ列挙している。評価項目については4.2節で詳細を示す。評価項目と技術レベルの交点には、当該の評価項目の技術レベルに対して期待される施策の例を記載している。これについては4.3節に例を示す。

#### 4.2 評価項目の策定

P, A, D, Sの4つの評価軸に対して、それぞれ10個を目途として合計30個の評価項目を策定した。評価項目は当初、PMBOK [11]の知識エリアと組込みスキル標準ETSS [12]の技術項目を参考に策定した。また、機能安全規格やセキュリティ規格との対応も意図している。その

表1 アセスメントシートの構成  
Table 1 Structure of the assessment sheet.

分類	ID	評価項目	技術レベル				
			Level5	Level4	Level3	Level2	Level1
開発プロセス (P)	P1	マネジメント技術	(評価項目・技術レベルごとに期待される施策を定義)				
	P2	開発プロセス設計					
	...	...					
アーキテクチャ (A)	A1	要求開発					
	A2	システムズ (群) アーキテクチャ					
	...	...					
設計・開発方式 (D)	D1	仕様記述・仕様確認					
	D2	テスト計画・テスト設計					
	...	...					
安心・安全 (S)	S1	セーフティ					
	S2	セキュリティ					
	S3	説明責任					

後、組み込みソフトウェアの概念拡大に伴う改訂を行っている。ソフトウェア技術あるいはソフトウェア開発技術の進展に伴い、求められる技術も変化する。特に、IoT (Internet of Things) や SoS (System of Systems) と呼ばれるつながるシステムの増加や、人工知能や機械学習技術を用いた自律するシステムの増加に対応している。たとえば、P11 不確実性への対応、A1 システムズ (群) アーキテクチャ、D11 不確実性への対応、S3 説明責任などが、近年のソフトウェアやソフトウェア開発に関わる社会環境の変化に伴って新たに設定した項目である。

以下に評価項目の一覧を示す。

- 開発プロセス (P)
  - P1 マネジメント体制
  - P2 開発プロセス設定
  - P3 開発環境マネジメント
  - P4 成果物マネジメント
  - P5 スコープマネジメント
  - P6 進捗マネジメント
  - P7 コストマネジメント
  - P8 品質マネジメント
  - P9 リスクマネジメント
  - P10 調達マネジメント
  - P11 不確実性への対応
- アーキテクチャ (A)
  - A0 要求開発
  - A1 システムズ (群) アーキテクチャ
  - A2 システムアーキテクチャ
  - A3 ソフトウェアアーキテクチャ
  - A4 ソフトウェア資産化
  - A5 ハードウェア/ソフトウェア分割
  - A6 ソフトウェアプラットフォーム
  - A7 ハードウェアプラットフォーム

- A8 ハードウェアコンポーネント
- A9 ソフトウェアプロダクトライン
- 設計・開発方式 (D)
  - D1 仕様記述・仕様確認
  - D2 テスト計画・テスト設計
  - D3 システム要求分析工程
  - D4 システム方式設計/ソフトウェア要求分析工程
  - D5 ソフト方式設計工程
  - D6 ソフト詳細設計工程
  - D7 コーディング工程
  - D8 ソフト単体テスト工程
  - D9 ソフト組み合わせテスト工程
  - D10 システムテスト工程
  - D11 ソフトウェア更新
- 安心・安全
  - S1 セーフティ
  - S2 セキュリティ
  - S3 説明責任

### 4.3 評価項目に対する技術レベルの定義

各評価項目に対して、技術レベルごとに要求される技術をリファレンス施策の例として提示している。先に示したように、Level1 から Level4 になるにしたがって要求レベルが上がる。下記に、P4 スコープマネジメントの例を挙げる。ただし、ノウハウ秘匿の観点において、実際のアセスメントシートから文言の一部を要約あるいは修正している。

- Level1 一部の成果物を集中管理
- Level2 開発中製品のソフトウェアのバージョンを組織的に管理
- Level3 製品や部署ごとに全成果物を関連付けて管理
- Level4 関連する製品群の間で、成果物の依存関係・流用

関係を管理

また、A1 システムズ（群）アーキテクチャに対する例を以下に示す。

- Level1 異分野組織をまたぐ価値を検討している
- Level2 最小価値のプロトタイプで市場価値を予測している
- Level3 異分野組織を含む全体価値と方針を定義している
- Level4 異分野組織をまたぐ全体の参照モデルがある

このように各評価項目の各レベルに対して、各組織で実践が求められる評価文言を定義することで、ソフトウェア設計力の評価精度を担保できるよう工夫している。

4.4 他標準との対応

先に述べたように評価項目と技術レベルは、他の標準との関係性を考慮しつつ、対象組織の組み込みソフトウェア開発プロジェクトに適用しやすいよう工夫をした。

4.4.1 評価軸の策定と ETSS の関係

開発プロセス（P）の項目策定において、ETSS の「管理技術」のスキル標準および PMBOK の知識エリアをベースとした。表 2 に ETSS と本アセスメントの開発プロセス（P）の関係を示す。ETSS の管理技術のうち「統合マネジメント」に代わり「マネジメント体制」を設定した（表 2 \*a）。策定当時の組み込みソフトウェア開発ではマネジメント業務が開発と未分離の場合もあったため、マネジメント体制確立の重要性を明示することを意図した。また、ETSS の管理技術のうち「組織マネジメント」と「知財マネジメント」は製品開発プロジェクトやソフト開発に閉じないことから、本強化施策の対象とせず別施策で扱うものとした（\*b）。ETSS の「コミュニケーションマネジメント」は情報共有等に関するが、「構成管理・変更

管理」に情報共有観点を融合した新たな評価項目として「成果物マネジメント」を設定した（\*c）。さらに、近年のニーズの流動性や技術変化の早さへ対応する能力として「不確実性への対応」を項目に加えている（\*d）。

設計・開発方式（D）の項目策定では ETSS の「開発技術」のスキル項目を参考としている。表 3 に ETSS の開発基準と本アセスメントの設計・開発方式（D）の関係を示す。ETSS では「システム方式設計」はシステム・エンジニアリング・プロセスに属し、「ソフトウェア要求分析」はソフトウェア・エンジニアリング・プロセスに属する。しかし、ハードウェアを中心に検討されることが多いシステム方式設計に対して、ソフトウェア開発関係者が関与しソフトウェア観点を考慮することが重要であると考え、その点を評価するようこれらを統合した（表 3 \*a）。「コーディング」と「テスト」は ETSS では 1 つのスキル項目となっているが、前者はコーディング規則への準拠や自動生成などの取り組みを評価し、後者は単体テスト方式の技術レベルを評価するものとして、別々の項目に分割した（\*b）。「システム結合」と「システム適格性確認テスト」は同一の工程で実施されることが多く統合した（\*c）。「ソフトウェア結合」と「ソフトウェア的確性確認テスト」も同様である（\*d）。「仕様記述・仕様確認」は独自の項目であり、モデルベース開発やモデル検査技術などの技術導入を明示的に評価するために追加した（\*e）。「テスト計画・テスト設計」は、これらが単体テストや結合テストなどのテストフェーズにかかわらず、共通して利用すべきスキルであると考え、独立した項目とした（\*f）。さらに、近年は OTA（Over the Air）技術などにより、組み込みソフトウェアにおいても出荷後のソフトウェア更新が重要となっていることから、「ソフトウェア更新」を追加した（\*g）。

表 2 ETSS 管理技術と開発プロセス（P）の関係

Table 2 Correspondence between the proposed management processes (P) and the ETSS management techniques.

ETSS 管理技術	開発プロセス (P)
統合マネジメント	P1 マネジメント体制 (*a)
スコープマネジメント	P5 スコープマネジメント
タイムマネジメント	P6 進捗マネジメント
コストマネジメント	P7 コストマネジメント
品質マネジメント	P8 品質マネジメント
組織マネジメント	— (*b)
コミュニケーションマネジメント	— (*c)
リスクマネジメント	P9 リスクマネジメント
調達マネジメント	P10 調達マネジメント
開発プロセスマネジメント	P2 開発プロセス設定
知財マネジメント	— (*b)
開発環境マネジメント	P3 開発環境マネジメント
構成管理・変更管理	— (*c)
—	P4 成果物マネジメント (*c)
—	P11 不確実性への対応 (*d)

表 3 ETSS 開発技術と設計・開発方式（D）の関係

Table 3 Correspondence between the proposed design and implementation methods (D) and the ETSS development techniques.

ETSS 開発技術	設計・開発方式 (D)
システム要求分析	D3 システム要求分析工程
システム設計	D4 システム方式設計
ソフトウェア要求分析	/ソフトウェア要求分析工程 (*a)
ソフトウェア方式設計	D5 ソフト方式設計工程
ソフトウェア詳細設計	D6 ソフト詳細設計工程
ソフトウェアコード作成とテスト	D7 コーディング工程 (*b) D8 ソフト単体テスト工程 (*b)
ソフトウェア結合	D9 ソフト組合せテスト工程 (*c)
ソフトウェア適格性確認テスト	
システム結合	D10 システムテスト工程 (*d)
システム適格性確認テスト	
—	D1 仕様記述・仕様確認 (*e)
—	D2 テスト計画・テスト設計 (*f)
—	D11 ソフトウェア更新 (*g)

4.4.2 独自の評価軸の策定

アーキテクチャ (A) は独自に策定した評価軸である。策定当初は、システムアーキテクチャ、ソフトウェアプラットフォーム、ハードウェアプラットフォーム、ソフトウェアコンポーネント、ハードウェアコンポーネントの5項目であったが、組み込み製品の機種展開が重要となり、要求開発とソフトウェアプロダクトラインを追加した。さらに近年には System of Systems 化を考慮し、システムズ (群) アーキテクチャ、ソフトウェア資産化、ハードウェア/ソフトウェア分割を追加した。

安心・安全 (S) も独自の評価軸である。セーフティとセキュリティについては各々に関する国際標準規格に準じた開発に至るステップをレベルとして示している。説明責任はそれらに共通して第三者への説明責任を果たす施策を示している。

ETSS のスキルカテゴリのうち「技術要素」は製品を構成する要素であるが、日立グループの組み込みソフトウェアを対象とする本施策では、製品ごとの構成要素の差異が大きいため、ETSS 技術要素に対応する項目は本アセスメントシートには含めていない。これは ETSS における技術要素の必要性を否定するものではない。ETSS は開発者個人を対象とするため開発製品の技術要素についてのスキルも重要である。本アセスメントは企業内の組織を対象とするため、その目的の違いにより、共通的な事項としてプロセス、アーキテクチャ、安心・安全を優先した結果である。

る。

なお、図1に示したスパイラルモデルについても、技術要素を利用して開発技術で開発してこれを管理技術で管理するという ETSS が想定する順とは異なり、開発プロセスを整備し、アーキテクチャを整えたいうで、適切な開発技術を導入すべきであるという順で考えている。これも製品開発に主眼のある ETSS と、組織の設計力を横断的に強化する本施策の目的の違いに起因する。

ETSS と本アセスメントはともに4段階 (Level0 を含むと5段階) であるが、ETSS では個人が技術革新を推進できる能力を評価するのに対し、本アセスメントでは組織横断的な強化施策の達成度という観点からレベル定義を表現している。また、ETSS ではキャリア標準において、職種や専門分野ごとに求められるスキル項目が示されているが、本アセスメントでは製品の技術特性やビジネス特性の違いを考慮した目標を各組織で設定することが特徴である。

4.4.3 他の標準との関係

SPICE [6], 機能安全規格 [8], セキュリティ規格 [9] とは、表4に示すような対応関係を想定している。ただし、本アセスメントの結果が各規格への準拠を保証するものではなく、各規格への準拠のためには、それぞれに定められたアセスメントが必要である。このような対応関係を持つことにより、事業上考慮すべき複数の規格への適合について、本アセスメントに基づいて俯瞰的に状況を把握できる。また、複数の規格への準拠を個別に推進すると類似し

表4 本アセスメントシートと他の標準規格との対応関係

Table 4 Correspondence between the proposed assessment sheet and the other standards.

規格	レベル	概要	本アセスメントシート			
			開発プロセス (P)	アーキテクチャ (A)	設計・開発方式 (D)	安心・安全 (S)
SPICE[6]	水準5	最適化	P1 Lv4, P11 Lv3	-	-	-
	水準4	予測可能なプロセス	P2 Lv4, P9 Lv4	-	-	-
	水準3	確立されたプロセス	P2 Lv3, P9 Lv3	-	-	-
	水準2	管理されたプロセス	P2 Lv2, P9 Lv2	-	-	-
	水準1	実施されたプロセス	P2 Lv1, P9 Lv1	-	-	-
	水準0	不完全なプロセス	-	-	-	-
機能安全規格 [8]	SIL4	仕様検査, 半形式手法	-	-	D1 Lv4	S1 Lv4
	SIL3	安全関連系と非安全系の分離, プロジェクト管理	P1 Lv3	-	D1 Lv4	S1 Lv3
	SIL2	構造化仕様, 構造化設計	P4 Lv3	-	D1 Lv3, D8 Lv2	S1 Lv2
	SIL1	仕様検査, 半形式手法	P4 Lv1	-	D4 Lv2	S7 Lv1
セキュリティ規格 [9]	EAL7	形式的検証済み設計及びテスト	-	-	D1 Lv4	S3 Lv4
	EAL6	準形式的検証済み設計及びテスト	-	A3 Lv4	D4 Lv4	S2 Lv4
	EAL5	準形式的設計及びテスト	-	A3 Lv4	D4 Lv4	S2 Lv4
	EAL4	方式的テスト及びレビュー	-	A3 Lv4	D1 Lv3	S2 Lv3
	EAL3	方式的テスト及びチェック	P8 Lv4, P9 Lv4	-	D9 Lv4	S2 Lv1
	EAL2	構造化テスト	P8 Lv3, P9 Lv3	-	D9 Lv3	-
	EAL1	機能テスト	P8 Lv1, P9 Lv1	-	D9 Lv2	-

た施策を重複して実施する必要性が出てくる場合があるが、統一的な設計力強化の枠組みで施策を選定・実践することで、組織として効率的な規格準拠を推進することができる。逆に、規格準拠が求められる製品開発においては、対応関係に基づき技術レベル目標を設定することも効果的である。表4にアセスメントシートとSPICE、機能安全規格、セキュリティ規格との対応表を示す。

#### 4.5 評価および目標設定

組織の設計力の評価はアセスメントシートの要求技術を参照し、それに該当する技術や施策を実践しているか否かを判定する。基本的には各レベルを達成しているか否かの二値（レベル評価の最低単位が1）での判定としているが、リファレンス技術の一部に着手している場合には0.5単位での評価をする場合もある。たとえば前節で例として示した「P4 成果物マネジメント」において、開発中の製品だけでなく当該部署内の過去製品や複数製品の管理がなされているもの、全製品あるいは全成果物とは言えない場合には、レベルをLevel2.5と判定する場合がある。

レベル評価を実施したのち、各組織や対象製品について達成したい目標レベルを要求項目ごとに設定する。これは、製品の特性やビジネスモデルなどにより、必要とされる技術が異なるためである。たとえば、「P4 成果物マネジメント」のLevel3における製品や部署ごとに全成果物を関連付けた管理は、工学的には実施されることが求められるものの、規模の小さなソフトウェアやバリエーションが少ない製品であれば優先度は低く、他の要求項目の改善を進めるほうがよい場合がある。このように、組織や製品の特性に応じた目標を相対的な優先度も踏まえて宣言することが本方式の特徴の1つであり、調達条件等としてのレベル

分類	ID	評価項目	対象外	レベル					目標	現状	GAP
				0	1	2	3	4			
P	P1	マネジメント体制							4	4	0
	P2	開発プロセス設定							4	3	1
	P3	開発環境マネジメント							4	3	1
	P4	成果物マネジメント							4	3	1
	P5	スコープマネジメント							3	3	0
	P6	進捗マネジメント							4	3	1
	P7	コストマネジメント							4	3	1
	P8	品質マネジメント							4	4	0
	P9	リスクマネジメント							4	3	1
	P10	調達マネジメント							3	2	1
	P11	不確実性への対応							4	3	1
A	A0	要求開発							4	3	1
	A1	システムズ(群)アーキテクチャ							4	3	1
	A2	システムアーキテクチャ							4	2	2
	A3	ソフトウェアアーキテクチャ							4	3	1
	A4	ソフトウェア資産化							4	3	1
	A5	HW/SW機能分割							4	3	1
	A6	ソフトウェアプラットフォーム							3	3	0
	A7	ハードウェアプラットフォーム							4	2	2
	A8	ハードウェアコンポーネント							3	3	0
	A9	ソフトウェアプラットフォーム							4	2	2
S	S1	セキュリティ							4	3	1
	S2	セキュリティ							4	3	1
	S3	説明責任							3	3	0
D	D1	仕様記述・仕様確認							4	4	0
	D2	テスト計画・テスト設計							4	2	2
	D3	システム要求分析工程							4	3	1
	D4	システム方式設計/ソフト要求分析工程							4	3	1
	D5	ソフト方式設計工程							4	3.5	0.5
	D6	ソフト詳細設計工程							3	3	0
	D7	コーディング工程							4	2	2
	D8	ソフト単体テスト工程							3	2	1
	D9	ソフト統合テスト工程							4	3	1
	D10	システムテスト工程							4	3	1
	D11	ソフトウェア更新							4	3	1

図2 レベル評価結果（値は説明用のダミー）

Fig. 2 Level evaluation results (values shown are dummy data for illustrative purpose).

達成を目的とする従来のアセスメント手法との違いである。レベル評価結果とそれに基づいて設定した目標レベルは図2のように一覧表で整理することで全体が把握できる。ただし、当該図は実際の組織を評価したものではない。目標と現状のギャップが視覚化され、ギャップが大きい項目から優先的に改善施策を実施することができる。

#### 4.6 設計力強化状況の把握

本手法によるアセスメント結果は、いくつかの観点で集計され状況把握に用いられる。

- (1) 特定製品に対する一時点での設計力と課題の可視化。評価観点ごとの評価結果と目標のギャップが用いられる。
- (2) 特定製品に対する経年による設計力の推移。強化施策の効果を把握する。このとき、評価項目ごとのレベルの変化を用いる。対象組織や製品により評価項目間で重要度に差があることに留意は必要だが、継続的な強化施策による設計力向上効果を俯瞰するには有効である。
- (3) 特定組織における製品ごとの設計力の相対比較。もしくは経年での組織の設計力の推移の把握。
- (4) 組織間での設計力と目標の相対比較。評価レベルの高低を単純に比較することは背景が異なるため意味がないが、類似した製品を擁する組織間では目標設定や施策実行の参考となる。
- (5) 企業グループ全体での設計力と目標設定の俯瞰。

### 5. ソフトウェア設計力評価・強化の推進

#### 5.1 推進体制

図3にソフトウェア設計力の評価と強化の施策にかかわる主な組織体制を示す。ソフトウェア設計力の評価および強化の対象となる組織を、本稿では「対象部門」と呼ぶことにする。また、ソフトウェア設計力の評価および強化の活動を先導する本社組織を「改革推進部門」と呼ぶことにする。改革推進部門は、グループ全体の組み込みソフトウェア設計力強化に関する責任を有し、対象組織群の品質

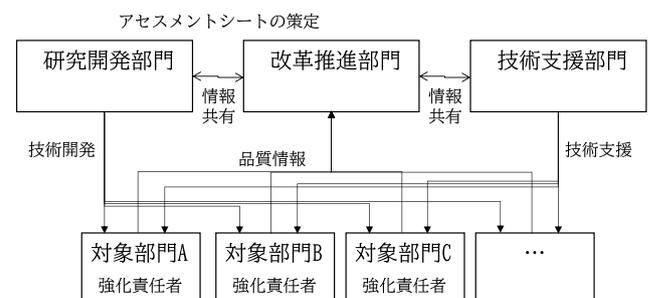


図3 設計力評価・強化の組織体制

Fig. 3 Organizational structure for evaluating and strengthening development capabilities.

情報を有している。また、ソフトウェア設計力の評価と改善に関する専門知識を有する。各対象部門には、それぞれ自組織内のソフトウェア設計力強化に責任をもつ「強化責任者」を置いている。改革推進部門の活動には、ソフトウェア工学に関する研究および対象組織への適用支援を推進している研究開発部門と、ソフトウェア開発に関する専門知識を有しプロジェクト管理やソフトウェア開発環境構築などで対象組織を直接的に支援する技術支援部門とが、企業グループ全体の状況を共有しながら連携している。改革推進部門と研究開発部門とでアセスメントシートの策定とメンテナンスを実施している。

### 5.2 設計力評価の位置づけ

本稿で示している設計力アセスメントシートの評価には時間と人的リソースを要し、企業グループが推進している多種多様多数のソフトウェア開発すべてに対して実践するのは現実的ではない。改革推進部門では、対象組織軍に対して定期的なヒアリングなどを行い、各対象組織のソフトウェア開発の体制や規模あるいは課題意識を把握している。これを概要アセスメントと呼んでいる。本稿で述べている設計力評価は、概要アセスメントにおいて改善が望まれる対象組織に対する詳細アセスメントの位置づけである。改善が望まれる対象組織は、製品品質等において課題がみられる組織、制御ソフトウェアからつながるソフトウェアなどへ製品の特性が変化している組織、開発規模や開発組織が増大している組織などである。

### 5.3 設計力評価の方法

本取り組みでは、設計力を評価する方法として下記のパターンを想定している。このように評価の方法に多様性をもたせることで、全体の評価コストを低減しながら、評価対象を増やす工夫をしている。

- (1) ワークショップ形式：対象組織と改革推進部門でワークショップを開催し、評価観点と技術リファレンスの解説を行いながら、参加者の合意に基づいて評価を行う
- (2) 事前説明+セルフアセスメント方式：改革推進部門から対象組織に対して評価観点と技術リファレンスの解説を行ったうえで、対象組織の関係者で評価を行う
- (3) セルフアセスメント方式：対象組織の関係者のみでアセスメントシートに基づいて評価する

## 6. セルフアセスメント環境

ワークショップ形式のアセスメントは第三者による客観的な評価ができる一方で、改革推進部門のリソースがアセスメント実施数の制約となる。グループ企業には数多くの組織において、さらに多くの製品やサービスの開発を行っており、それらすべてについてワークショップ形式のアセ

スメントをすることは現実的ではない。そこで、セルフアセスメントを効率的に行うための Web ベースのアセスメント環境を構築した。本セルフアセスメント環境では、アセスメントシートに基づくセルフアセスメントを対話的に実施できることに加え、特定の評価項目に対する強化を行う際に参考となるプラクティスを参照することで、設計力強化施策まで効果的に策定することを意図している。

### 6.1 セルフアセスメント環境による設計力評価

セルフアセスメント環境では、評価項目ごとのページで評価項目の説明と観点を参照しながら、目標レベルと現状レベルを入力できるようになっている。図 4 に、セルフアセスメント環境におけるレベル入力画面の例を示す。目標レベルと現状レベルのほか、その評価の根拠や、次のレベルに至るために実践すべきアクションなどをコメントとして記載できるようになっている。

さらに、過去に実践されたレベルごとの事例を関連プラクティスとして参照することができる。関連プラクティスは次のレベルに到達するための施策の一例である。関連プラクティスを具体的に示すことは、各レベルの定義を理解するための参考情報にもなっており、組織横断的に設計力を評価する一助ともなっている。関連プラクティスは、関連する評価項目とレベル、プラクティスの名称、目的、概要、適用の前提条件、社外事例、参考文献が記載されている。図 5 に関連プラクティスを表示した画面の例を示す。(画面の一部を墨消している)

関連プラクティスの項目の意味は次のようである。  
**目的** プラクティスによって達成すべき事項を示す。設計力評価指標における評価項目ごとの達成レベルに応じた目標が示される。  
**概要** プラクティスで行う内容の概略を示す。



図 4 セルフアセスメント環境の画面例 (評価項目; 値は説明用のダミー)

Fig. 4 Example of self-assessment interface: evaluation item screen (values shown are dummy data).

## #127 アジャイル開発(スプリント)

♡ CP×20    いいね! ×    コメント ×    添付ファイル ×  
 Knowledge Level2 P11:本業実務への対応 公開  
 が が に投稿 が が に編集 <更新履歴>

### アジャイル開発(スプリント)

#### 目的

- 価値あるソフトウェアを素早く継続的に提供する
- 提供ソフトウェアに対するフィードバックに素早く対応する
- 顧客要求の変更/追加に対応する

#### 概要

- 計画的反復型開発の実践
  - 各段階でクリアすべき開発項目の明示
  - 積み残し項目を管理(バーンダウンチャート)
- ステークホルダ分析/顧客を巻き込んだ体制づくり
  - プロダクトオーナーの指名
  - バックログ管理
  - 契約
- プロトタイプ開発/フィードバック獲得(F2F)
  - 曖昧・動的な要件獲得
  - 不確実性への対応

#### 適用の前提条件

図5 セルファアセスメント環境の画面例(関連プラクティス)

Fig. 5 Example of self-assessment interface: related practices screen.

**前提条件** プラクティスのための前提条件を示す。プラクティス間には依存関係がある。当該プラクティスの前段となるプラクティスをこの項目に明示する。

**事例** プラクティスの本体である事例を示す。その実態は、過去改善事例の説明資料やソフト開発のガイドなどを参照する。また、独自に準備した成果物テンプレートも事例として提供している。

**参考文献** プラクティスを理解/実践するうえで参考となる書籍や Web 等を示す。

## 6.2 評価結果の表示

すべての評価項目について評価を完了すると、図6に示すように評価レベルと目標レベルを俯瞰することができる。このとき、評価レベルと目標レベルの乖離が大きい評価項目をハイライトすることで、強化すべき項目を把握できるように工夫している。各評価項目に対して求められるレベルは組織や製品ごとに異なるため、評価レベルが低い項目を強化することが必要ではなく、設定した目標に対して評価が低い項目の強化が必要であることに注意が必要である。

なお、レベルは定性的評価の結果であり、数値が定量的な意味を持つものではない。たとえば Level1 と Level2 の差は、Level2 と Level3 の差と同等ではない。この点で図6のようなグラフ表現は認知の誤りを招くリスクがあるが、管理者が全体感を把握するための参考情報として採用しているものである。実際の運用では、数値の差をもって強化点を決定するのではなく、定性的な評価内容を考慮し

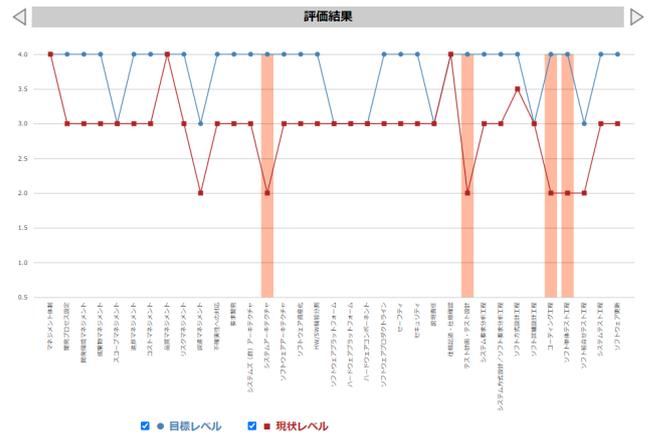


図6 セルファアセスメント環境の画面例(値は説明用のダミー)

Fig. 6 Example of self-assessment interface: evaluation results screen (values shown are dummy data).

推奨プラクティス		
	次レベル	現状レベル
マネジメント体制		SEPG・PMOの推進手続
開発プロセス設定	プロセスの継続的改善	開発プロセスKPI定義/把握
開発環境マネジメント	InnerSource	開発環境/ツールの集中管理/サービス化
成果物マネジメント	製品系列での資産管理	成果物/レーサビリティ管理
スコープマネジメント	アジャイル開発(スプリント)	要求変更に伴う影響分析
	プログラムマネジメント	チケット情報補正

図7 セルファアセスメント環境の画面例(推奨プラクティス)

Fig. 7 Example of self-assessment interface: recommended practices screen.

て対策を検討している。

## 6.3 強化施策の策定支援

先に示した関連プラクティスが評価項目のレベルごとに蓄積されているため、あるレベルから次のレベルに上がるために実施するプラクティスの例として関連プラクティスを利用することができる。そこで、図7にあるように、各評価項目のレベルに該当する関連プラクティスを「現状レベル」で実施していることが期待されるプラクティスとして、評価レベルの次のレベルに該当する関連プラクティスを「次レベル」で実践するプラクティスの候補として表示するようにしている。これにより、現状レベルの確認、あるいは現状レベルでの異なるプラクティスの気づき、次レベルに向上するための強化施策の策定を支援している。なお、関連プラクティスは企業ノウハウであるため、本稿では一部のみを開示している。

## 7. 適用事例

本章では日立グループにおける組み込みソフトウェア設計力評価の実践事例を示す。ただし、目標レベル、評価レベルの具体値については企業機密であるため、相対的な結果についてのみ示す。

7.1 対象事例

2019年4月から2021年6月に実施された評価結果について述べる。この間に実施された評価の対象は、12の事業部門または事業会社の25製品である。うち、1事業会社2製品は国外の事例である。対象となるソフトウェアの多くは機器を制御する組み込みソフトウェアであるが、ネットワーク接続などの情報処理を含むものもあり、機器から得られるデータを可視化したり遠隔で指示を行ったりする情報ソリューションも含まれる。

7.2 評価レベルおよび目標レベルの分析

本節では、評価レベル、目標レベル、それらのギャップについて、組織間のバラつきに着目したアセスメント結果の分析結果を示す。ここで製品間での評価レベルと目標レベルのバラつきにより、当該項目の相対的な変化状況を推測できると仮定する。

製品間で評価と目標のいずれも特定のレベルに集中している項目は、重要性の変化は小さく、実際の設計力レベルも均等であると考えられる。逆に評価レベルと目標レベルの両方について製品間でバラつきがある項目では、評価と目標の差のバラつきが小さければ、各製品で求められるレベルに対して必要な設計力を各組織が備えていると考えられるが、評価と目標の差のバラつきが大きければ、先行して要求が変化している製品で設計力強化が追い付いていないことが推察される。また、評価レベルがバラつき、目標レベルが特定のレベルに集中している場合には、当該項目の設計力に組織間の差異が大きく、評価レベルが特定のレベルに集中し目標レベルがバラついている場合には、一部の製品で高い目標を求められる新しい要求があるものと推察される。後者の場合、評価と目標の差が小さい製品で

も、今後より高い目標設定が必要となる可能性が示唆される。すなわち、製品や製品に求められる項目の変化を経年変化だけでなく、状況の異なる複数製品のバラつきから推察できると考えている。以上を、表5にてまとめる。

表6に、4つの評価軸のそれぞれについて、レベルがバラついている評価軸を示す。リスクマネジメントの評価レベルが突出して製品間でバラつきが大きい。これは製品ごとにリスクの考え方が異なることが反映されているものと考えられる。近年は、製品の成熟により不確実性の低い開発と、ネットワークなどの新たな技術要素の導入により不確実性の高い開発があり、リスクマネジメントに対する取り組みが先行している製品とそれほどでもない製品の差が開いていることが示唆される。

このように、個々の製品では開発期間が長く、評価項目の重要性の変化や設計力強化の効果の把握には長期間を要するが、変化スピードが異なる多数の製品の状況を比較することで、製品特性や求められる評価項目の変化を推測できる。これは全社共通施策として強化を要する項目の選定に有用である。

8. 議論

本稿で示したアセスメントによる評価と目標設定による設計力強化のプロセスは、IDEALモデル[13]を踏襲したものである。すなわち、アセスメントシートによる設計力評価はIDEALモデルにおけるDiagnosing(診断)フェーズ、目標設定はEstablishing(確立)フェーズ、強化施策の実行はActing(行動)フェーズに相当する。一方、Initiating(開始)フェーズについてIDEALモデルでは改善活動の着手時に一度実施することとなっているが、製品や製品を取り囲む環境あるいは開発組織などが頻繁に変化する

表5 評価レベル、目標レベル、ギャップの分散による状況推測

Table 5 Status inference based on distribution of evaluated levels, target levels, and gaps between them.

評価レベルのバラつき	目標レベルのバラつき	目標と評価ギャップのバラつき	バラつきから推測される状況
小	小	-	変化が少なく全社を通して一定の設計力を確保できている
大	大	小	製品による差が大きいが各製品に対して必要な設計力を確保できている
大	大	大	一部の製品で重要性が大きく変化しており、対応に差がある
大	小	-	製品間での設計力にバラつきがある
小	大	-	一部の製品で重要性が大きく変化しており、新たな要求への対応が求められる

表6 製品間のバラつきが大きい評価項目

Table 6 Evaluation items with large variability in results across products.

評価の軸	評価レベル	目標レベル	ギャップ
開発プロセス (P)	リスクマネジメント	成果物マネジメント	調達マネジメント
アーキテクチャ (A)	ハードウェアプラットフォーム	ソフトウェアプラットフォーム	ソフトウェアアーキテクチャ
安心・安全 (S)	説明責任	セキュリティ	説明責任
設計・開発方式 (D)	テスト計画・テスト設計	ソフト方式設計工程	コーディング工程

る現代においては、一度の開始で継続的な改善を進められるわけではない。そのため、5章に示したように、企業グループ横断的な体制を整備しながら、定期的な概要アセスメントにより変化をとらえて開始フェーズを再実施することが望ましいと考える。また、IDEALモデルのLearning(学習)フェーズでは行動フェーズの妥当性を評価し将来活動を提案することを求めているが、筆者らの取り組みでは単一組織での分析に加えて、複数組織の評価結果や目標設定それに基づく改善施策を共有することで、次の活動方針の策定を支援していることも特徴である。

本稿で示した設計力評価の仕組みの特徴の1つは、評価レベルを一律に向上するのではなく、各組織や製品で評価レベルに基づいて設定した目標レベルをめざして設計力の向上を図る点にある。そのため、評価と目標のギャップに着目した仕組みづくりを推進している。

表6のギャップのバラつきに着目すると、安心・安全の説明責任と開発プロセスの調達マネジメントでレベルが分散している。ギャップが分散している評価項目は、製品や社会の変化が大きい分野で先行して目標レベルが上がる傾向にあり、それほど変化が小さくなくギャップが小さい分野との差が開いたものと想定される。つまり、ギャップがバラついている項目は特に近年になって重要性が高まっている項目であると推察される。そのように考えると、各ソフトウェア製品が社会に与える影響が大きくなり、社会との関わりが重要になる状況があり、説明責任が重要になってきているものと推察される。また、調達マネジメントは従前でも重要であるが変化が少ない製品については適切な仕組みが構築されていると考えられるが、ソフトウェアの機能が増え、また自社で閉じた製品開発からOSSや導入品を用いた製品開発に変化することにより、新たな調達マネジメントの課題が表出化しているものと思われる。このように、レベルとギャップのバラつきに着目した分析により、組み込みソフトウェア開発を取り囲む状況の変化を見出す方法の1つとなることが期待されるが、その妥当性の検証については今後の課題である。

## 9. まとめ

本稿では、企業グループ横断的な組み込みソフトウェア設計力評価の枠組みとその実践について述べた。多種多様な事業や製品を抱える企業グループにおいてソフトウェア設計力を評価する共通の枠組みとして、開発プロセス、アーキテクチャ、安心・安全、設計・開発技術の4つの軸で整理したアセスメントシートを策定し、製品ごとのレベル評価を実践している。また、評価されるレベルそのものではなく、評価レベルに基づいて各組織・製品で設定する目標に対する強化施策を推進している。この点が従来のSPICEなどの成熟度レベルの向上を目標とするアセスメントとは異なる点である。また、これらの施策を推進する

ための組織体制や、セルフアセスメントを効率的に実践して取り組みの拡大を図るセルフアセスメント環境について述べた。さらに、実際に評価・目標設定を行った複数製品の結果に対して、組織間のバラつきを用いて組み込みソフトウェア開発の状況の変化を読み取る試みを示した。今後はこれらをベースとした組み込みソフトウェア設計力強化の取り組みを継続するとともに、状況変化を見越したプロアクティブな仕組みへと発展させていきたいと考えている。

## 参考文献

- [1] 大原茂之：組み込みソフト産業の実態と開発の課題：日本の組み込みシステム産業と技術者育成の課題，情報処理，Vol.46, No.2, pp.161-168 (2005)。
- [2] 小泉 忍，鍵政豊彦，川口 進，菊池 淳：日立グループシナジーを生かした組み込みシステム開発力強化の取り組み，日立評論，Vol.91, No.5, pp.418-419 (2009)。
- [3] 小川秀人：企業におけるソフトウェア開発とソフトウェア工学，コンピュータソフトウェア，Vol.20, No.3, pp.2-11 (2011)。
- [4] 小川秀人，川上真澄，加賀洋渡，長船辰昭：日立グループ本社におけるソフトウェア開発プロセス革新，日立評論，Vol.101, No.2, pp.76-81 (2019)。
- [5] ISO/IEC 15504: 2003: Information technology – Process assessment, Standard, International Organization for Standardization (2003)。
- [6] ISO/IEC 33020: 2015: Information technology – Process assessment – Process measurement framework for assessment of process capability, Standard, International Organization for Standardization (2015)。
- [7] VDA QMC Working Group 13 / Automotive SIG: Automotive SPICE Process Assessment / Reference Model, Standard (2017)。
- [8] IEC61508: Functional safety of electrical/electronic/programmable electronic safety-related systems, Standard, International Organization for Standardization (2010)。
- [9] ISO/IEC 15408: Information technology – Security techniques – Evaluation criteria for IT security, Standard, International Organization for Standardization (2009)。
- [10] 金子昌永，石川 誠，森 拓郎：医療機器ソフトウェア開発を対象とした包括的アセスメントのケーススタディ，ソフトウェアエンジニアリングシンポジウム2020論文集，情報処理学会，pp.206-210 (2022)。
- [11] Project Management Institute: *A Guide to the Project Management Body of Knowledge* (2021)。
- [12] 情報処理推進機構：組み込みスキル標準 ETSS2008 (2008)。
- [13] Carnegie Mellon Software Engineering Institute: *The IDEAL Model*, Technical report (2009)。



小川 秀人 (正会員)

1996年名古屋大学大学院博士前期課程修了。同年株式会社日立製作所入社。現在、同社研究開発グループ主管研究長。ソフトウェア工学の研究と適用支援に従事。2015年北陸先端科学技術大学院大学博士後期課程修了。博士(情報科学)。2023年より北陸先端科学技術大学院大学産学官連携客員教授。情報処理学会, 電子情報通信学会, 日本ソフトウェア科学会 各会員。



島袋 潤 (正会員)

1990年大阪大学大学院博士前期課程修了。同年株式会社日立製作所入社。研究開発グループを経て、現在はプラットフォームサービス事業部門に所属。一貫して日立グループ内でのソフトウェア工学の実用化に取り組む。技術士(情報工学)。情報処理学会会員。



日野 雅透

1999年徳島大学大学院工学研究科修士課程修了。同年株式会社日立製作所入社。現在は本社品質保証部門に所属し、ソフトウェア開発力・信頼性強化活動に従事。



若林 昇

2001年神戸大学大学院自然科学研究科修士課程修了。同年、株式会社日立製作所入社。本稿執筆時は同社研究開発グループ主任研究員。組み込みソフトの生産技術に関する研究に従事。



川上 真澄 (正会員)

1998年、豊橋技術科学大学大学院工学研究科修士課程修了。同年株式会社日立製作所入社。2023年、電気通信大学情報理工学研究科博士課程修了。現在、同社研究開発グループ部長。ソフトウェア工学研究と製品開発適用に従事。IEEE, 情報処理学会, プロジェクトマネジメント学会各会員。