

# システムズエンジニアリング導入実施の一事例 報告書

2018年3月1日

---

## 目次

はじめに.....	3
1. システムズエンジニアリングの普及に取り組む背景.....	4
1.1 ソフトウェア開発の環境変化 .....	4
1.2 システム開発の新しいアプローチ・システムズエンジニアリング .....	5
1.3 システムズエンジニアリングのポイント説明.....	5
2. パイロット活動の目的・対象・活動方針.....	7
2.1 パイロット活動の動機・目的.....	7
2.2 対象システムの選定 .....	7
2.3 対象システムの概要 .....	8
2.4 パイロット活動の課題と対応方針.....	9
3. 実践内容.....	11
3.1 取り組みと成果 .....	11
3.2 活動のまとめ.....	18
4. 今回の活動の一般化 .....	20
4.1 パイロット活動から得た活動のヒント .....	20
4.2 システムズエンジニアリングについての考察 .....	22
4.3 参考情報 .....	24
おわりに .....	25
謝辞.....	26

---

## はじめに

近年、IoTの進展により、製品／サービスが多様化しネットワークにつながるようになったことを背景に、これまでにない組み合わせでモノ・コトをつなげて、高い価値をもたらすことが期待されている。その一方、対象となる分野・範囲の広がり等により、要件は複雑化し、多くの専門分野をまたいで統合する広い知見も求められるようになってきている。製品／サービスの企画・開発を行おうとする際、特定分野に特化した従来型のアプローチそのままでは目的の達成は難しくなっている。

このようなシステムの企画・開発における課題の効果的かつ包括的な解決方法として、既に宇宙・航空分野を中心に実績があり、欧米では有効性が認知されて一般的な製品やシステムへの適用が進んでいるのが「システムズエンジニアリング」である。IPA/SECでは、このアプローチが日本においても有効と考え、普及・展開活動に取り組んでいる。[1][2]

本書は、IPA/SECが三菱重工機械システム株式会社(以下 MHI-MS社)のITS\*事業部門と協力して行ったシステムズエンジニアリングのパイロット活動について報告するものである。この活動では、“実証実験プロジェクト”におけるプロトタイプシステム開発の上流設計工程において、システムズエンジニアリングの考え方を導入する取り組みを行った。(\*ITS: Intelligent Transport Systems)

この活動を通して、システムズエンジニアリングの考え方がより一般的なシステム開発においても有効であることを確認すると共に、実開発現場の課題への地に足の着いた対応を促進するための実際の開発プロジェクトに向けた知見を紹介する。

今回のパイロット活動の対象であるシステム開発は以下の条件に該当しており、新しいアプローチが必要になるタイミングであった。同様の環境にある方々には、特に参考にしていただきたい。

- 従来、単品の製品を開発し、一定の成功は収めてきたが、その製品を含めた付加価値の高いサービスを実現しようとする場合
- 要件が決まればきっちり作る自信はあるが、自らの技術、製品を取り巻く環境を一段高い視点から分析しなければならなくなった場合

---

## 1. システムズエンジニアリングの普及に取り組む背景

### 1.1 ソフトウェア開発の環境変化

#### (1) 多くの機器のつながり

「ソフトウェアの開発」を本業として進めてきた組織・部門にとって、昨今のビジネスのトレンドである IoT の活用、例えば「安価なセンサなどの大量の汎用機器がつながるシステム」の企画・開発等に携わる機会が増えている。これらのシステムを作り上げ利用するためには、つながるすべての機器を含めたシステム全体を対象に、その設置から使用開始後の運用・維持管理、さらには廃棄のことも考慮して取り組まなければならない。

#### (2) 広がるシステム同士のつながり

自らが開発もしくは管理する対象には「大量の機器」などが存在しなくても、「外部の様々な機器・システムと連携して自らの機能・サービスを実現するシステム」に関わる機会が増えている。つながることでの大きな価値提供をビジネス好機と捉えることを意識して、様々な他者との連携を考慮し、全体を見通して取り組まなければならない。

#### (3) 要求を導出することの難しさ

(1)と(2)で述べたつながりが時間と共に広がっていくという環境下では、システム開発に関して、実現性があり、かつ目的の達成に向かうように要求事項を導出することが難しくなっている。最初に必要な変化の可能性を見通して要求事項をすべて抽出することは難しい。

大きな方向性を見据えつつも、「段階的に進める」アイデアを出して早く動き出し、具体的なフィードバックを得ることでより明確な要求事項を見出ししていく、そんなアプローチも必要になっている。

#### (4) 品質の考え方への影響

多様性と、時間的変化の影響で、品質要求の適切な水準を決めることも難しくなっている。

こうした環境では時間が経過するうちに、様々なシステム同士がつながることによる品質の不整合が発生しうる。例えば、それまでの品質水準が大きく異なっていたものたちが相互につながるケースである。特に高品質を必要としないソフトウェア(例:突発的に使えなくなることがあっても大きな問題ではない、レスポンス性能に切迫した条件がない、等)として開発期間短縮重視で作ったものが、あるときから業務システムに接続することも考えられる。業務の遂行に欠かせないミッションクリティカルなシステムの一部を担うことを期待され、開発当初に想定しなかった高品質(例:突発的に使えなくなることがあってはならない、レスポンス性能に切迫した条件がある、等)を求められることも起こり得るのである。

#### (5) ライフサイクルを意識する重要性の高まり

IoT 時代のシステムが継続的に価値を発揮し続けるためには、環境変化に耐え、発展的に対応

---

していくことが求められている。そして、システムがその期待に応えるためには、変化を見込んだ柔軟性を持っておくことが必要であり、そのためには当初からシステムライフサイクルを見通して設計しておく必要がある。

当初に見極め切れない要素があれば、それを徐々に明確化するようにステップを踏んで進めながら、ライフサイクルを通して担うべき責任範囲をコントロールしていくことが、ビジネスとしての成功のために必要である。

## 1.2 システム開発の新しいアプローチ・システムズエンジニアリング

1.1 節で紹介した様々な環境変化に耐えられないシステムは、期待される自身の価値を維持できないことで、つながっている大きなシステムの価値まで毀損しかねない。

こうした課題認識から、ビジネスとしてつながった大きなシステムの価値を得るためには、システムを俯瞰的に捉えるアプローチが有効だと考えられる。

そうした取り組みのアプローチとして期待されているのがシステムズエンジニアリングである。

システムズエンジニアリングとは、システムを成功させるための複数の専門分野にまたがるアプローチと手段である。ここで言う「システム」は、コンピュータシステムにとどまらず、機械、電気機器、人間系（操作者）、環境など広い意味を持っている。

システムズエンジニアリングは、航空・宇宙等の領域で長年にわたって培われてきた企画・開発のアプローチを汎用的に体系化したものである。ソフトウェアシステムやハードウェアシステムだけでなく、新規事業の開発、社会システムの設計など、様々な領域に適用可能である。

システムズエンジニアリングの効能としては、以下がある。

- 本来の目的を明確化し、より高い視点や幅広い知見を取りまとめてアプローチする方法なので、新たなビジネスチャンスの把握やリスクの明確化が可能である。
- 関係者が共通的に理解できるような概念整理を行い、繰り返しブラッシュアップすることにより、環境変化に耐えうる、より良い製品／サービスを生み出すことができる。

システムズエンジニアリングは、その成り立ちにおいては高度な専門性を要する複雑なシステム開発に適用され、そこで培われたノウハウを体系化したものである。前述のような様々な環境変化に直面している IoT システムなどの一般的なシステム開発においても、有効性を発揮できると想定され、円滑な導入を進めていくことが期待されている。

## 1.3 システムズエンジニアリングのポイント説明

システムズエンジニアリングの重要な考え方として「4つのポイント」を示す。システムズエンジニアリングの特徴の中で、主な課題に対応しており、経営層を含め関係者に広く理解しやすいポイントとして取り上げたものである。

#### 4つのポイント

- 目的指向と全体俯瞰
- 多様な専門分野を統合
- 抽象化・モデル化
- 反復による発見と進化

(出典:「経営者のためのシステムズエンジニアリング導入の薦め」 [2])



図1 システムズエンジニアリングの4つのポイント

これらのポイントを中心に、2017年に新しいアプローチの有効性を伝えるための2つの啓発書を公開した。[2]

「経営者のためのシステムズエンジニアリング導入の薦め」

「開発者のためのシステムズエンジニアリング導入の薦め」

経営者向けの啓発書では、システムズエンジニアリングの適用事例や有効性を紹介し、開発者向け啓発書では更に、導入に向けたポイントや、活かすための考え方を説明している。

---

## 2. パイロット活動の目的・対象・活動方針

### 2.1 パイロット活動の動機・目的

多くのシステム開発の現場では、それぞれ従来のポリシーや規則、開発標準などがあり、それらが QCD の目標達成に寄与している。それらに織り込まれた考え方や手法を、いきなりすべて捨てて開発の手法をあらためる、というのは現実的でない。円滑に開発現場への導入を促進するためには、開発現場の環境条件を考慮する必要がある。

開発現場はそれぞれ培ってきた知見(専門性やノウハウ)を保有している。新たな IoT システムは、なんの知見の蓄積もないまま取り組むのではなく、保有した知見を強みとして発展的に取り組むのが一般的である。こうした開発現場では、直面する多様性や複雑性の課題に対処するためにシステムズエンジニアリングを学習して適用する場合、保有していた知見と新たなシステムズエンジニアリングに関する知見との整合性を考慮しながら、現実的に地に足の着いた対応を促進する必要がある。従って、システムズエンジニアリングの一般的な紹介や解説を行うだけでは、現場課題への対応を促進する上では不十分である。開発現場への適用に向けて、実際の開発プロジェクトでの活動を通じた知見やノウハウの獲得が重要であると考えた。

実活動から得られる知見を通してシステムズエンジニアリングの考え方を導入するノウハウを体得すると共に、報告書として取りまとめ、業界内で広く知見を共有することを目的として、パイロット活動に着手した。

### 2.2 対象システムの選定

一般のシステム開発に関わる中で、システムズエンジニアリングの考え方を取り入れることが有効な対象として想定している状況は以下のようなものである。

- ・従来、単品の製品を開発し、一定の成功は収めてきたが、その製品を含めた付加価値の高いサービスを実現しようとしている
- ・要件が決まればきっちり作る自信はあるが、自らの技術、製品を取り巻く環境を一段高い視点から分析しなければならなくなっている

これらに該当する開発を実践の場とすることで、多くの一般的なシステム開発の参考になる知見が得られると考え、いくつかの開発プロジェクト候補を見据えた結果、MHI-MS 社の ITS 開発が最もシステムズエンジニアリング適用に相応しいと考え、パイロット活動の対象として選定した。

## 2.3 対象システムの概要

### (1) パイロット活動の対象システム

MHI-MS 社は ITS(高度道路情報システム)分野において実績があり、新たな案件のプロトタイプシステムの開発と実証実験を計画していた。そのシステムの中核を担うサーバ上の情報システムを開発するチームにおいて、システムズエンジニアリング導入のパイロット活動を実施することとなった。

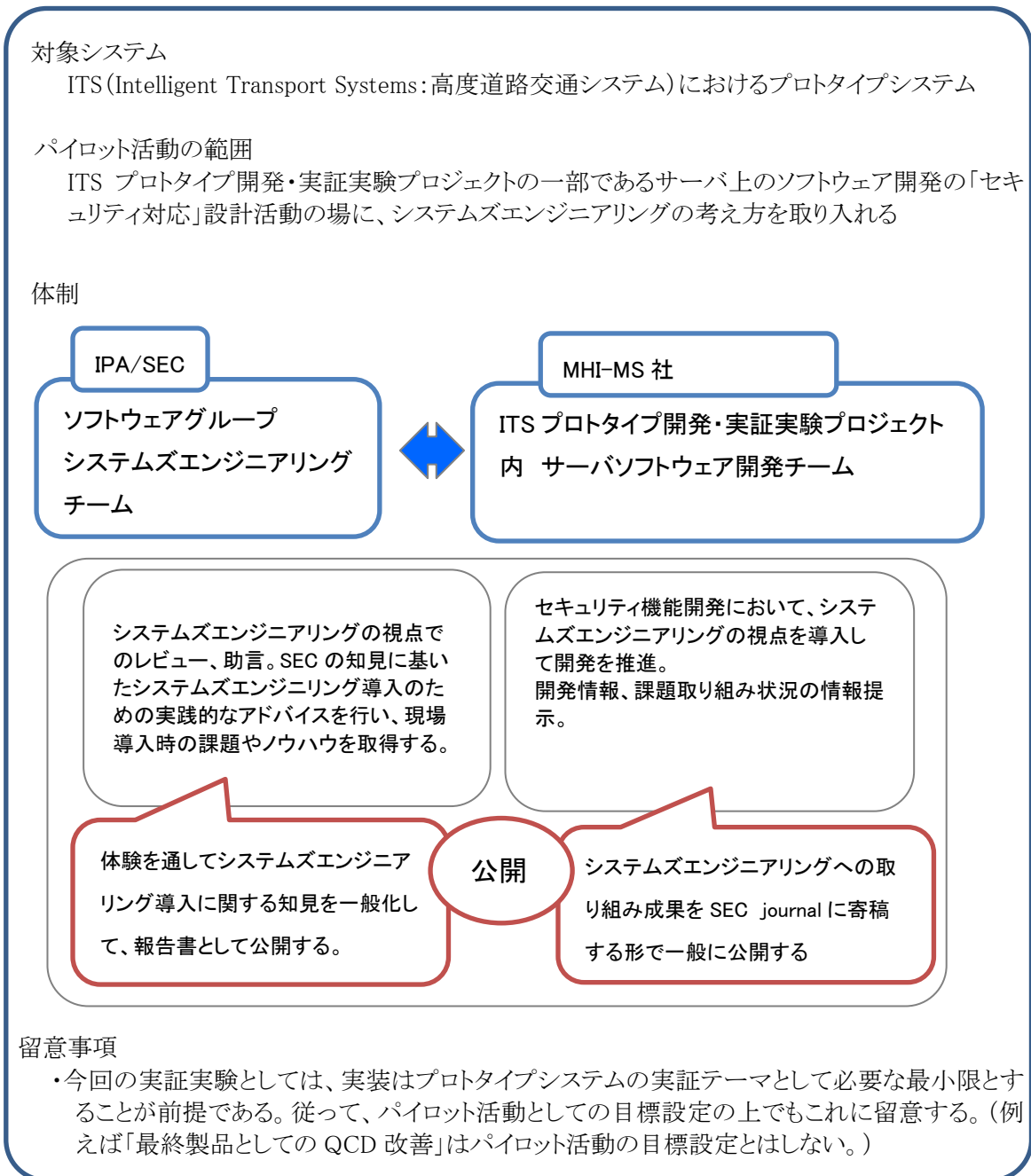


図 2 パイロット活動の環境まとめ



---

## (2) 対象システムの課題とシステムズエンジニアリングの考え方への期待

ITS 事業の領域においては、技術の進歩とサービス向上の追求に伴いシステムの形態が変化してきている。道路交通関係のシステムとしては、例えば ETC ゲートのような専用機器システムがよく知られている。しかし、今後はそうした専用機器にとどまらず、カメラ/センサなどの汎用機器の活用も前提になってきている。それらを路上など屋外に設置し、オープンなネットワークを介して情報のやり取りを行うことで、従来よりも低コストで、かつ、今後の画像技術、センサ技術、通信技術等の進歩を取り入れやすい柔軟なインフラの構築が、競争の上で重要となっていた。

今回の活動の対象は、そうした新たな価値を実現するシステムについてプロトタイピングでの実証実験を行うプロジェクトであった。

そしてサーバの情報システムにおける今回のプロトタイピングでの主要なテーマのひとつがセキュリティ対応、特にオープンネットワーク接続に対応して外部からの攻撃に対して安全な対策を講じること、であった。従来取り組んできた固定された環境の中にある製品・サービスでは設計根拠が希薄であった「セキュリティ要件」に、新たに対応する必要性が生じていたのである。

セキュリティは一過性の対処で解決できる課題ではない。攻撃を受けるリスクは継続的に存在し、攻撃内容は今後ますます巧妙化することが予想されている。従って、新しい種類の脅威が発見されることを想定して、変化への対応を継続していくことを見込んでおかなければならない。

また、進化の著しい「ITS 分野のビジネス」ということで、スタート時点では決定できない将来的なビジネス面の拡大を配慮した柔軟性、拡張性が必要である。セキュリティ要件についても、将来的な第三者サービスとの連携など、データの取り扱い要件が変わる際には、合わせて変更されることが想定される。セキュリティ技術にのみに焦点を当てるのではなく、ビジネスやシステム全体との整合性が常に求められる。

これらの課題認識に基づき、直面するセキュリティ対応の機能の取り込みのみならず、この機会にライフサイクルのレンジで捉えたシステムズエンジニアリングの考え方を取り入れて、プロトタイピング～実証実験の活動の成果を向上させることが重要と考えた。こうして、新しいアプローチを試行すべく、パイロット活動に取り組むこととした。

パイロット活動の対象としてはセキュリティ要件に関わる開発としているが、システムズエンジニアリングの導入としては、一部機能に限定した開発への適用でなく、より広い範囲のシステム開発としての将来的な適用を念頭に置いている。

## 2.4 パイロット活動の課題と対応方針

### (1) パイロット活動としての課題

活動を行う部門には、これまでのシステム開発で利用してきた開発標準があり、それらに従って開発を推進している。パイロット活動の着手に先駆けて、対象のサーバソフトウェア開発チームとしての設計作業は既に始まっていた。開発の範囲としては、セキュリティ対応以外にも案件があり、並行開発する他チームとの整合性の面でも、速やかに設計を進めていく必要があり、設計資料など

---

---

の開発ドキュメントは定められている開発標準に従って作業を始めていた。

開発標準はソフトウェアライフサイクルプロセス標準を基に、自部門のシステム開発などの知見を総合して作られたものであり、これまでの開発で一定の成果を挙げてきていた。

こうした、部門を支えてきたこれまでのやり方と整合をとって、システムズエンジニアリングの考え方を導入して、かつ、有効性を発揮することが今回の課題であった。

## (2) パイロット活動の対応方針

新たなことに取り組み有効な結果を体験するには、学習と実践が必要である。システムとして捉える考え方や体系化されたシステムライフサイクルプロセスの知識といった基礎的な学習を行うことと並行し、具体的な活動を通して効果を実感することが重要と考えた。そして、有効な活動にするための適切なポイントを見出すために、現状の開発標準のプロセスをシステムズエンジニアリングに関わりの深い主要なプロセスと比較して、付加的に価値を発揮できる実践のポイントを見出すこととした。ここで比較対象は IPA/SEC でのシステムズエンジニアリングの WG 活動を通して「主要な 7 プロセス」を抽出済みであったので、その 7 プロセスに絞って行うこととした。そして、見出したポイントについて実践に移すこととした。(プロセスについての詳細は 3.1(2)を参照)

また、システムズエンジニアリングの考え方を実践する効能は一過性のものではない。今回のパイロット活動から得られた知見を今後の開発に活かすことが重要である。本来のシステム実現に向けて必要となる「取り組みすべき事項」や「改善すべき内容」を明文化して、今後の開発(あるいは次の段階のプロトタイピング)に反映できるように知見として伝えることも、本パイロット活動の一部として取り組み、アウトプット作成を目標とすることとした。

---

## 3. 実践内容

### 3.1 取り組みと成果

前述の方針に基づき、大きく以下の4つの活動に取り組んだ。

システムとして捉える考え方の学習に相当するのが①、体系化されたシステムライフサイクルプロセスの知識の学習と開発標準との比較が②、実践に相当するのが③、次に向けて伝えることのために相当するのが④である。

- ① 基本的考え方の理解：システムとして捉え「目的指向と全体俯瞰」を徹底
- ② 有用な知識の習得：「システムライフサイクルのプロセス」の理解と開発標準との比較
- ③ 開発における実践：「上流工程での妥当性確認の実践等」と成果の実感
- ④ 今後の開発に向けた申し送りのまとめ：「パイロット活動の取り組みの整理」

そして、今後の開発に向けて「システム構築手法説明書」を作成して、重要事項をノウハウや課題として明文化して残すことをパイロット活動のゴールとして設定した。

上記の①～④の取り組みは必ずしも番号順に行ったものではなく、並行して実施し、かつ、一回ではなく繰り返し行ったものである。

- ・基本的な考え方として「目的指向と全体俯瞰」の重要性について、実際の開発の課題に関わる情報交換、意見交換を繰り返し、共通理解を図った。(①)
- ・システムライフサイクルプロセスの標準(ISO/IEC/IEEE 15288:2015) [3] から、特にシステムズエンジニアリングに対応する特徴のある7つのプロセスを学習して、開発標準との比較を行った。そして、実践に移すポイントを抽出して、その意義の共通理解を醸成した。(②)
- ・開発作業の一部として、比較によって抽出したポイントについて実践した。一例として「妥当性確認」を設計書レビューとして実際に行い、システムズエンジニアリングの考え方を導入することの効果を確認した。(③)
- ・上記の活動を通して全体感を掴みながら、開発として取り組みすべきことやその状況などを俯瞰して、「今回見直したこと」、「今回は見直さなかったが今後に向けて見直すべきこと」などを整理した。これらの内容は、今後の開発に向けて課題や実践すべき事項を誤解なく漏れなく申し送りするドキュメント「システム構築手法説明書」としてまとめた。これが今回のパイロット活動が今後の開発に寄与するアウトプットである。(④)

それぞれの取り組み内容を以下で説明する。

#### (1) 基本的考え方の理解

パイロット活動の最初の段階で、前述の「システムズエンジニアリングの4つのポイント」の中でも最も基本に位置する「目的指向と全体俯瞰」について、重要性の理解を図った。

今回の対象システムは、道路上のカメラなどの機器と情報システムがオープンなネットワークを介してつながる一種のIoTシステムである。今回のサーバ開発チームは、そういう機器を含ん

---

---

だシステムとしての開発に関しては経験知が豊富であったが、あえて、サーバ以外の機器やネットワーク、運用後の対応など、サーバ上の開発の範疇外で考慮が漏れそうなことを指摘することで、確認することとした。指摘は空間的な網羅性、時間的な網羅性、意味的な網羅性の観点から行った。

そこで、例として以下のような事項が挙げられた。

「道路設置機器まで通したセキュリティの考慮」

(道路に設置された機器、ネットワークも含めたセキュリティ脅威への対応が必要)

「セキュリティリスクの進化、巧妙化への継続的対応」

(運用期間を通してのセキュリティ脅威への対応が必要)

「ビジネス面の拡張への対応」

(新たな要件により外部システムとの間で個人情報データを共有するような変更が生じ、当初のセキュリティ要求事項にも影響が生じることへの対応が必要)

図 3 ディスカッションの中で挙げられた考慮すべき事項の例

ここで列挙した例はいずれも、今回のプロトタイプでは限定対応にとどめるが、最終的なサービスリリースまでには対応もしくは拡張性としての考慮が必要な事項であった。つまり、将来的に大きな手戻りが発生しないように配慮しておくべき基本的な要件に関わる事項であり、単に残項目として記録しておけば良いというものではなかった。

A) ありがちな整理レベル:

割り切り事項を記録としてまとめて課題リストとする

(何か考えなければいけないことのリストとして残る)

B) 全体を見通した計画レベル:

「今回の対処の理由や制限事項への対応アイデアなどの考え方」を記録して申し送る  
(次の開発への申し送り事項として活きる)

図 4 残った課題の整理のレベル

残項目の整理のレベルを図に示した。対応のスピードやフットワーク優先で作業を進めていると、レベル A の整理に陥りがちである。実際にそういう整理でも問題がないような些末なことも多々存在する。一方、今回のような実証実験ではなく実際に稼働するシステムとしては影響が大きく、レベル B の整理を行っておくべき事項もある。上に列挙した件はいずれもレベル B に該当する。こういう整理のメリハリは現実には製品／サービスの目的を理解し、全体を俯瞰していないと適切な見極めはできない。

ここで紹介した内容は一例であるが、こういう実際の課題に関する討議を通して、「目的指向と全体俯瞰」というポイントを意識することが、適切な設計に大きく影響することについての共通理解を深めた。

## (2) 有用な知識の習得

システムズエンジニアリングの基礎知識として、システムライフサイクルプロセス (ISO/IEC/IEEE15288:2015) [3]を学習すると共に、部門の開発標準との比較を行った。なお、部門の開発標準のベースになっていたのは、ソフトウェアライフサイクルプロセス[4]であった。

システムズエンジニアリングのベースであるシステムライフサイクルプロセスのうちの7つのプロセスについて、従来のソフトウェアライフサイクルの考え方と比較して特徴があるので、それらに焦点を当てて学習、及び実作業を進めることが、この部門の開発に新しい価値を加えるために有効と考えた。

対象の7プロセスを以下の図に赤字と下線で示す。

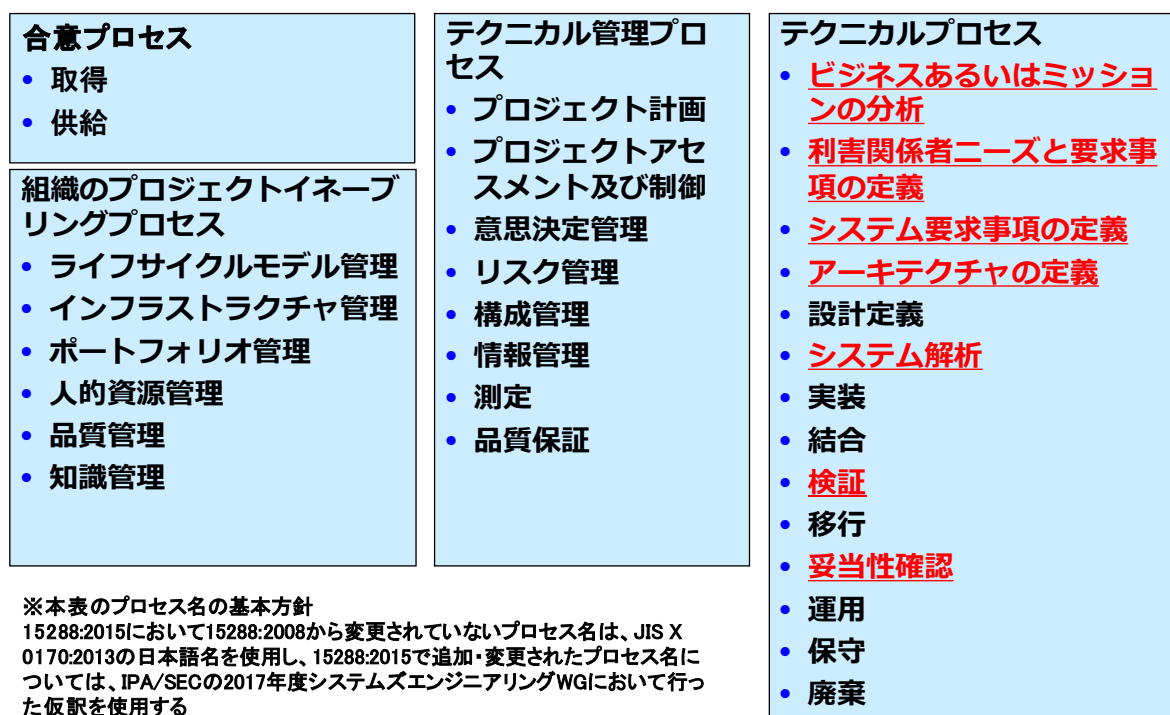


図5 システムライフサイクルの技術プロセス概観と7つプロセス

部門で持っている開発標準について、上記の「システムライフサイクルプロセス」からの7つのプロセスに関わる内容との比較を行って、不足しているプロセスなどの洗い出しを行った。

なお、大きな差異を見極めるべく、プロセス単位での比較とし、アクティビティ、タスクレベルの詳しい比較は行わなかった。

その比較の結果、開発標準の特徴として以下の内容が明らかになった。

- i. ビジネス面も含めた「システムの目的」を明確にするプロセスが規定されていない。具体的には以下のプロセスに相当する内容は規定に含まれない。
  - ビジネスあるいはミッションの分析
  - 利害関係者ニーズと要求事項の定義

- 
- ii. 上流で考慮すべき範囲として、移行、稼働後の運用、維持への対応については規定されていない。
  - iii. 上流の各工程での妥当性確認を行うことが明確に規定されていない。

こうした特徴がある理由としては、i については、企画・営業活動として実施して開発標準の範囲としてはまとめてこなかったためであると考えられる。ii、iiiについては、従来の当該部門のビジネス範囲であれば暗黙のルールなどがありベテランの有識者の対応で自ずとカバーされ、明文化の必要が希薄であったためと考えられる。

しかし、これまでに経験してこなかった領域の取り組みをするにあたっては、従来の企画・営業部門やベテランの知見に頼って、これまでのような高品質な開発を行える保証はない。

こうした、比較と考察を通して、開発現場から見たパイロット活動として、

- ✓ ビジネス企画の段階での技術的な検討の実施
- ✓ 上流での移行・運用など開発後に必要な事項の配慮
- ✓ 開発の上流工程における妥当性確認の実施

の3点を今回の取り組みで注力する対象の「候補」として洗い出すことができた。

### (3) 開発における実践

(2)で洗い出した3点の候補の中から、プロトタイプ開発で効果が見えやすい「開発の上流工程における妥当性確認」を実践した内容を紹介する。実際の開発作業にシステムズエンジニアリングの考え方を導入して「従来と何が異なるのか」を実体験した内容である。

開発標準では、妥当性確認は最終検証におけるものとして明記されていた。設計段階のレビューは、前工程からのインプットに基づく設計作業の検証作業に相当するものであって、システムの目的に対する妥当性確認というレビューは明記されていなかったのである。

そこで、設計書のレビューとして新たに「妥当性確認」を目的としたレビューを追加して実施することにしたのである。

設計書の妥当性確認レビューは以下の通り実施した。

対象: ユースケース(機能仕様の設計ドキュメントのうちセキュリティに関わる部分)

レビュー方法: 査読レビュー

観点: インプットとなった機能概要資料だけではなく、目的(要件検討時の資料に基づく)に照らして、目的と設計内容の整合性のレビューを行う

トピックス : セキュリティに関するユースケースとして、人手による作業とフィールド機器との通信にフォーカスしてレビューを行った

---

このレビューを通して、ITSの専門家でない第三者であるIPA/SECのメンバーから、本来の目的に照らした「ユースケースで記載された内容に関する疑問点」や「より考慮すべき点」などの指摘事項を出すことができた。当然専門知識が伴わないため内容的には拙いところが多々ある指摘であるが、妥当性確認の観点を意識する姿勢を形として表した。特に、複数の動作仕様間の整合性に関する疑問点から、「本来の目的」と「決められた動作仕様」の整合性の確認につながるような指摘は、専門家でなくても設計仕様書の記載内容から可能な指摘であった。

指摘内容は結果的に、障害としてすぐに修正するのものと、プロトタイピングの今回は手を付けずに次の段階への課題として残す扱いとしたものがあつた。(課題リスト掲載 12 件)

長く経験を積み上げてきた従来の開発であれば、設計時に分解した機能をきっちり実装して検証しながら組み上げれば、概ね全体の目的を達成することができていたので、強いて上流で「妥当性確認」を強調しなくても大きな問題が出なかったと思われる。

しかし、新たな環境での開発において、本来の目的との乖離してしまうことを避けるには、常に目的指向で確認していくことが有効である。そのことを実感する活動であつた。

なお、妥当性確認の実施は、開発標準のプロセス定義にいきなり手を付けるのではなく、レビュー観点としての導入から試行できるので、始めやすい取り組みである。ただし、考え方の理解が伴わないと有効な実践につながらないので、その点の留意が必要である。

#### (4) 実践を通じた気づき

妥当性確認の活動を通して、ベテラン有識者の知見の有効性に関する気づきがあつた。活動に関わっていただいた有識者によると、妥当性確認に相当するレビューの観点は、従来の開発では、視野を広く持ったベテランによりカバーされていたことが多い、ということであつた。予想はしていたことであつたが、具体的な指摘内容からあらためてそのことが確認されたのである。開発標準にプロセスとして明記されていなくても、ベテランの有識者からは「妥当性確認」に相当する指摘もされていたということである。長年携わってきた特定の枠組みの開発においては、固有のノウハウが現場に蓄積され、結果的にカバーされていたものと考えられる。しかし、そのノウハウが一般化されておらず、結果として「妥当性確認」を意図的に上流で実施するプロセスは定義されていなかったということである。

実績のある開発チームの貴重なノウハウが、個人の「暗黙知」の形でのみ存在し、実施内容やその理由・根拠が組織に明示的に共有されていなければ、新たな環境での開発において継承し、応用して活かされるとは限らない。

今回の活動は、「上流工程での妥当性確認」ということを通して、担当者の世代交代も踏まえて、有識者の暗黙知の有効性や、それを引き出すには、標準的なシステムライフサイクルプロセスの参照が有用であるということを確認することにつながつた。

・妥当性確認についての補足

妥当性確認の上流工程での実践については、システムズエンジニアリングならではの知見ではなく、ソフトウェア開発の指針の中でも述べられてきた内容である。下図はその代表的なものとして、「共通フレーム 2013」[5]において示されている内容である。

システムズエンジニアリングを導入するという取り組みにより、それぞれの開発の特性に応じてテーラリングにより取り込まれなかったプロセスについて、再検討、再評価を行うきっかけとなる。これも取り組みの効能のひとつである。

【背景】V&V(Verification and Validation)の適用場面を示し、何と何を比較するのかを明確にし、読者の方々に理解・実践してもらうことにより、システム及びソフトウェアの品質向上を図る。

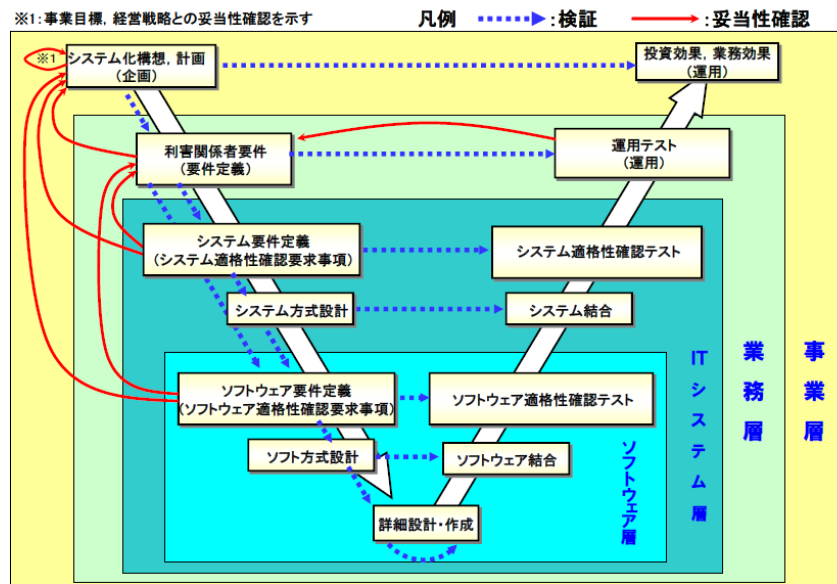


図4-xx 検証(Verification)と妥当性確認(Validation)の適用場面

※ 2013/3/7 IPA/SEC プロセス改善セミナー資料「共通フレーム 2013 の概要」(P58) を基に作成

図 6 妥当性確認の適用場面



(付記)

システムライフサイクルプロセスと部門開発標準の比較から抽出した候補の残り2点についての扱いについて簡単に説明する。

✓ ビジネス企画の段階での技術的な検討

今回のパイロット活動の開始時には既にプロトタイプ開発プロジェクトとしての企画段階を過ぎていたので、今後の開発において企画段階での技術的評価を加えることを、パイロット活動からの申し送りとして整理した。

パイロット活動の実践としては、目的指向と全体俯瞰に関する検討の過程で、ビジネス上の課題を起点としてセキュリティ設計開始につなげるべく、プロトタイプシステム開発方針や範囲の整理をするためのプロジェクト要素分析シートを作成して対象の分析を行った。

また、今後につなげるべく、設計完了に近い時点での振り返りの活動として、目的展開図などを使用してプロジェクトの本来の目的と実施策の整理を、あらためて実施した。

✓ 上流での移行・運用など開発後の配慮

今回のプロトタイプ開発においては、移行、運用・維持の実施を簡略化し、次回開発における課題として「システム構築手法説明書」への記載として残すこととした。

## (5) 今後の開発に向けた申し送りのまとめ

今回の学習と実践内容から、今後の開発に向けて開発プロセスの改善を行うべく、「システム構築手法説明書」に記述して申し送り事項として残すこととした。本格開発時への申し送り事項であり、開発標準に補強するポイントの候補としても一般的に意義のある資料として残すこととした。(本活動は MHI-MS にて開発活動の一環として実施)

なお、ベテラン有識者の知見が暗黙知となっており、これを含めて明文化していくことが有意義であることは、ある程度予想はしていたが、当初の計画時点での確信はなく、活動の過程で気づきとして得ることができた。上記のプロセスの比較を通してみることでベテラン有識者の暗黙知がどういう点を補っており、有効であるかということを一時的プロセスに当てはめて確認することができたことは有益であった。

今回の活動をきっかけに、新たな IoT 時代を迎えて、組織として保有している知見の重要性が認知されたのである。

---

## 3.2 活動のまとめ

### (1) 直接の成果

今回は、パイロット活動として、主に目的指向と全体俯瞰の観点で対象を捉えなおすことで、開発チームが自ら開発の新たな取り組みを導入する活動につながった。

システムズエンジニアリングの取り組みとしては入り口に立った状態であり、システムのライフサイクルを考えた開発への見直しのスタートラインに立ったところである。具体的な成果としては、上流での妥当性確認を行うという開発慣習の改善に加え、今後のシステム開発では企画段階から技術的評価を加えることを、今回の活動からの申し送りとしており、実際の運用・維持を考慮した設計の必要性も課題として申し送られている。

そして、標準的なシステムライフサイクルのプロセスを参照することで、有識者の暗黙知に多くの改善のヒントがあるであろうことにも気づくことができた。それぞれの活動を通して、更なる改善の視点を、開発チーム自身が持って、続けていただくことができると考えている。

今回のアウトプットである「システム構築手法説明書」がその成果である。これは、セキュリティ対応だけに閉じることなく、より視野を広げて、サーバのソフトウェア開発チームとしての開発に、今回の「システム構築手法説明書」を活用していくものである。

### (2) パイロット活動から得られた気づき

一般的なシステム開発にシステムズエンジニアリングの考え方を導入する活動を通しての気づき事項を記載する。

まず、部門が持つ開発標準とシステムズエンジニアリングのアプローチの整合性をどのように図っていくかという課題を解決するために、開発標準とシステムライフサイクルプロセス標準の代表的プロセスとの比較を行うことが有効であった。具体的に、妥当性確認に関する充実を図ることが重要であることなど、今後の開発に重要な気づきを得ることができた。

また、開発標準には記載されていないノウハウを主にベテランの有識者が暗黙知として保有しており、上手く引き出せば効果的に活用していくことができる可能性にも気づくことができた。プロセスの比較の結果を元に、システムズエンジニアリングと対応付けることで、そのベテラン有識者の暗黙知の意味や有効性が理解しやすい形になるので、それらのノウハウをドキュメント化して残すことができる。暗黙知を引き出して共有するためのひとつの方法として、開発標準とシステムライフサイクルプロセス標準の比較が有効であることを知ることができた。

### (3) システムズエンジニアリングの効能

なお、本書で紹介しなかった内容も含め、システムズエンジニアリングの4つのポイントやシステムライフサイクルの良い点を活用するために、全体の開発作業を俯瞰して整理を行い、有用と思われることについて一部着手した。大まかな内容を以下に図示する。

	ポイント				
	目的指向と全体俯瞰	多様な専門分野を統合	抽象化・モデル化	反復による発見と進化	その他
テクニカルプロセス (6.4.1)【ビジネスあるいはミッションの分析】 (6.4.2)【利害関係者ニーズと要求事項の定義】 (6.4.3)【システム要求事項の定義】 (6.4.4)【アーキテクチャの定義】 (6.4.6)【システム解析】 (6.4.9)【検証】 (6.4.11)【妥当性確認】 その他	設計の上流と捉えて、分析シートによりビジネスとプロトタイプ開発対象を整理 ↓ 上流における妥当性確認レビュー実施	部門開発標準の適用 海外の交通システムに関する知見（法律面を含め）など広範な領域を束ねるプロジェクトとして体制構築して推進	SQuaRE品質モデルでセキュリティで要件を整理 ↑ 部門開発標準の適用	部門開発標準の適用 プロトタイプPRJを推進	経験あるエンジニアの暗黙知に依存 ↓ 開発プロセスとして明文化（継続）
	従来は企画・営業活動の段階として、開発プロセスとしていない 経験あるエンジニアの暗黙知に依存 → 課題として抽出し明文化				

図 7 ポイント×プロセス対応マップと今回の対策

この図からシステムズエンジニアリングの考え方を導入した活動を振り返ってみると、以下のよう  
なことがわかる。

開発の各プロセスに4つのポイントをマッピングしてみると、「多様な専門分野の統合」については体制面を含め考慮して推進されており、「反復による発見と進化」の考え方は既に取り入れてプロトタイプングに取り組んでいた。今回の取り組みでは、主に「目的指向と全体俯瞰」のポイントでの追加対策と、これまで暗黙知の領域であったところを明文化することにつながっている点の特徴であることが、この図からわかる。

既に実施されていることも含めて対応範囲が可視化され、不足している対応などを考える上で有効な整理となった。なお、これは漏れ防止のチェックシートではない。すべての欄に必ず何か新たな対策をするという必要はなく、あくまでも整理のためのシートであることに注意されたい。

---

## 4. 今回の活動の一般化

### 4.1 パイロット活動から得た活動のヒント

システムズエンジニアリングが自社の業務に役立ちそうだと思ったら、どうすれば良いだろうか。

本パイロット活動と類似した環境にある場合は、システムズエンジニアリング導入を軸とした改善のアプローチとして、本パイロット活動は参考になるとと思われる。本章では、パイロット活動の事例を一般化した上で、システムズエンジニアリングに取り組もうと思った人がそれに立ち向かうヒントを例示する。

#### (1) パイロット活動の振り返り

今回のパイロット活動をシステムズエンジニアリングの考え方を利用した改善活動としてまとめると以下ようになる。

##### 「現場の課題」

クローズド環境の専用システムからオープンな環境での汎用的な技術を使用した新たな製品サービスに取り組みを広げるにあり、従来の開発アプローチだけでできるのか漠然とした不安があり、システムズエンジニアリング導入の取り組み開始を判断。

↓

##### 「システムズエンジニアリング導入に向けた現場の課題」

いきなり大きく異なる開発手法の導入は難しく、既存の開発標準に整合した開発の推進が必要であり、システムズエンジニアリングを導入するにしても小さく付加的に実施して効果を確かめるような工夫が必要である。

↓

##### 「その克服のための有効だった進め方」

システムライフサイクルプロセスをベースとして現状の開発標準を調査することで、現実的な実施策を見出した。本パイロット活動では、「設計書の妥当性確認レビュー」がそれに該当する一例である。

↓

##### 「得られた成果」

- ・妥当性確認レビューにより具体的な改善指摘が抽出された
- ・妥当性確認レビューを設計に組み入れるよう見直しがされている
- ・目的指向と全体俯瞰の考え方の有用性について気づきを得て 企画段階の技術評価、運用・維持の考慮 について課題として認識され、次の開発に申し送りされている
- ・高スキル保有者(有識者)の知見(やっていたこととその理由・根拠)を伝承することの有用性について気づきを得た

↓

##### 「システムズエンジニアリングの考え方の導入について獲得した知見」

標準的なシステムライフサイクルの主要プロセスを参照して開発標準を見直すことで、重要なことの漏れと有識者の暗黙知で支えられていた点に気づくことができた。

---

## (2) 今後の導入開始へのヒント

今回のアプローチから一般的に導入時点のアプローチとして有効であると思われる要素を紹介する。

### ① システムライフサイクルプロセスをベースとしたアプローチ

対象：実績のある開発標準を持っている組織、プロジェクトなど

内容：「システムライフサイクルプロセス」の7プロセス(SEC が選定した7プロセス)を参照し、自部門の開発標準と照合(\*1)して意味を考える。そして7プロセスを比較対象として、自部門の有識者の暗黙知を整理、評価して、新たな開発標準の運用に供するようドキュメント化する。

\*1 ここでいう照合は、プロセスの単位で大きな比較を行い、差異を明らかにする。この際、細部の差異は追及しない。そして細部まで調べないとわからない一貫性や差異は網羅できなくてもやむを得ないものとする。これらは導入後に適宜改善することとする。  
自部門の開発が標準化されていない、開発標準が形骸化して実態は標準化されていない、などの状況においては本書で紹介している事例の知見を活用することは難しい。ソフトウェアエンジニアリングの初歩から取り組みされることを推奨する。

期待効果：効果的な活動着手のポイントが発見できる。

### ② (参考)上流工程での妥当性確認の実施

本来は①のアプローチの後に続くものであるが、本件は短期に試行することを想定して参考として記載する。

対象：ソフトウェア開発プロジェクトで主に設計、開発、検証に関わる人

内容：設計書の妥当性確認レビューを行う。

期待効果：本来の目的と設計が乖離することを防止できる。(問題の早期発見)

特に、本件は、開発標準にないプロセスを追加するような大がかりな変更のアプローチではなく、レビュー観点を追加して実施するような工夫で着手できるので、試行としてもやりやすいものである。(なお、例えば、レビュー計画の周到な準備が開発標準で規定されている場合は、準備の作業の一環として「検出すべき問題として妥当性確認に関わる観点を含めておく」必要がある。開発標準に準拠しながら適用していくことが望ましい。)

内容的には、個々の機能細部に拘ったレビューとは一線を引き、「本来の目的」と「決められた仕様」の整合性に拘る視点で行う。糸口としては、複数の仕様間の整合性や、仕様の詳細化にあたり、開発の都合に流されない意識を持つことが核となる。

---

## 4.2 システムズエンジニアリングについての考察

本書で紹介した例は、新たな IoT 環境に手を広げる開発部門である。従来のノウハウで培った開発標準の枠組みを積極的に見直して、より高い成果を上げる開発の仕方に踏み出すために、システムズエンジニアリングという言葉掲げシステム全体を対象に思考するということを明文化して取り組んだ結果である。

協力いただいた開発部門の目的は成果の向上と開発力の向上であり、システムズエンジニアリングはそのためのアプローチとして試行し、評価したものである、システムズエンジニアリングが目的ではない。

今回のパイロット活動では、システムライフサイクルプロセスの活用が有効と考え、取り組みを進めた。システムズエンジニアリングとしては、アプローチの導入に向けた第一歩であるということであらためて述べておく。

なお、ともすれば、システムズエンジニアリングを既存のソフトウェア開発の方法論に対峙する方法論のように捉え(誤解し)、有効性が証明されるまで採用に値しないというような意見も見受けられる。一方、今回紹介した、システムライフサイクルプロセスの参照による開発アプローチの見直しは、決してこれまでのソフトウェア開発方法論の否定ではなく、環境に応じた最適化の見直しを推奨しているものであって至極あたりまえの取り組みともいえる。

システムズエンジニアリングとして体系化された膨大な知見[6]については、それぞれの開発部門の特性に応じた取り入れ方が必要であるが、共通のベースとしてシステムライフサイクルプロセス[3]を参照することを意識いただき、取り組みされることをお勧めする。

以下に今回の取り組みで、システムライフサイクルプロセスを参照したのが、システムズエンジニアリングを目的としていたからであろうという誤解を避けるべく、システムライフサイクルを参照するに至った思考の流れを記載する。一般的な状況として記載しているが、今回も該当した内容である。

ご参考にされたい。

ソフトウェアの開発現場はそれぞれの製品・サービスの開発を通して培った開発標準を持っている。一般的に多いのは ソフトウェアライフサイクルプロセス[4] を参考に、固有の条件等に応じてテーラリング(必要なものを足し、必要性の薄いものを除き、より有効な運用に向けて内容を具体化するなど)し、自分たちの開発に最適化して、それに基づいて開発を行っている。

**現在、IoT のような新しい取り組みに直面した人たちは、従来の対象システムに向けた最適化の結果である現在の開発標準は必ずしも今後に向けた”最適”ではないという現実**に気づいている。

新たな取り組みにあたって、自身のこれまでの製品・サービスで培った土台をベースにつながりを広げていくようなビジネスケースでは、自身の強みを活かしつつ、新しい分野にも広げていくことになる。従来の開発標準をすべていきなり捨てて新しいアプローチの0から考える、というのでは、強みすら毀損するような弊害(例:品質保証の機能が働かないなど)が想定され、現実的でない。

既存の稼働資産のような守るべきものがない、あるいは、既存の品質を毀損するリスクを容認できる、などのケースでは、

「すべて捨てて新しいことを導入できるはずだということを前提に、0から新しいやり方の  
試行錯誤を繰り返す」

という選択もありうる。しかし、こういう前提もなく0から始めるのは論理的でなく、危険な選択である。

本書では、既存品質の毀損が容認できないケースを対象としている。

強みを活かしつつ発展的に見直していくのであれば、自身の強みであった開発標準を、新環境に向けて再定義することが必要になる。その手始めとしては、まず、新しいビジネスの領域について調べ、考え抜くことが当然必要であるが、それに続いて、開発としてやるべきことも変わってくるので、既存の開発標準に欠けていること、余分なことを見直していくことになる。

この際、思い付きの範囲の見直しにとどまらないよう、一般の標準を参照して比較してみると有効である。

特に、部門の開発標準はこれまでの最適化の過程で、一般のソフトウェアライフサイクルの標準から除いたり、変更したりしたものがあるので、それをあらためて見直してみると有効な第一歩である。

ただし、IoT などの新たな環境には、IoT の多様な機器や関係者の関わりの広がりなどソフトウェア以外の要素が大きく、ソフトウェアライフサイクルの標準の範囲ではカバーするのは難しい。より広くシステムとして捉えた開発の推進に結びつけるために、今回の取り組みではシステムライフサイクルの標準[3]がより有効であると考え、それをベースとした見直しを実践した。

---

### 4.3 参考情報

参考情報を記載する。

- [1]「ドイツ・欧州企業におけるシステムズエンジニアリングの実践に関する調査・分析結果報告」  
及び「ドイツ・欧州企業におけるシステムズエンジニアリング 実践課題とベストプラクティス」

<https://www.ipa.go.jp/sec/reports/20161219.html>

- [2]「経営者のためのシステムズエンジニアリング導入の薦め」

及び

「開発者のためのシステムズエンジニアリング導入の薦め」

<https://www.ipa.go.jp/sec/reports/20170329.html>

- [3] ISO/IEC/IEEE 15288:2015 Systems and software engineering – System life cycle processes

- [4] ISO/IEC 12207:2008 Systems and software engineering – Software life cycle processes

- [5] 共通フレーム 2013 ～経営者、業務部門とともに取組む「使える」システムの実現～、  
(IPA/SEC)

- [6] Guide to the Systems Engineering Body of Knowledge (SEBoK)

[http://sebokwiki.org/wiki/Guide\\_to\\_the\\_Systems\\_Engineering\\_Body\\_of\\_Knowledge\\_\(SEBoK\)](http://sebokwiki.org/wiki/Guide_to_the_Systems_Engineering_Body_of_Knowledge_(SEBoK))



---

## おわりに

多くのシステム開発現場が新たな環境変化に直面しており、これに立ち向かうべくシステムズエンジニアリングのアプローチの導入を検討するケースが増えてくると想定されている。本書はそうした取り組みの際に、それぞれ持つ知見(専門性やノウハウ)を強みとして活かしながら新しいアプローチを導入するための、実用的で参考になる情報を紹介した。

従来の枠から一歩踏み出して新しいビジネスに乗り出すためのシステム開発に関わるすべての人に向けて、従来の形だけにとらわれることなく、より広い視野をもって問題解決に取り組むことの重要性を、システムズエンジニアリングの学習を通して理解していただければ幸いである。本書が、現状への問題意識を持った多くの方たちにとって、新しいアプローチへの取り組みを開始するきっかけになることを期待する。

---

## 謝辞

本パイロット活動にあたり、ご協力いただいた皆様、特に貴重な活動を通してフィードバックを実施していただいた三菱重工業株式会社、及び三菱重工機械システム株式会社 の関係者の皆様に深く感謝の意を表す。