「システム思考」ガイドブック（入門編）

視点を拡げ、多くの視点を持って考えよう

平成 30 年 6 月
ロボット革命イニシアティブ協議会
IoT による製造ビジネス変革 WG
System of Systems 調査研究チーム

目次

はじめに 3
1. 時代背景の再認識 4
2. 本書の狙い 5
2.1. 想定する対象の読者 5
2.2. 読者に伝えたいこと 5
2.3. 本書の位置づけ 5
2.4. 対象とするシステム 6
3. システムに関する定義 7
3.1. 既存のシステムの定義 7
3.2. システム関連用語の整理（既存の定義） 7
3.2.1. システム科学 7
3.2.2. システム・アプローチ 7
3.2.3. システム指向 7
3.2.4. システム・セオリー 7
3.2.5. システムズ・エンジニアリング 7
3.2.6. システム・オブ・システムズ 8
3.2.7. システム・デザイン 8
4. 身近な事例 9
4.1. 実装優先開発 9
4.2. 暗黙知開発 9
4.3. 変化を好まない変化への対応 9
4.4. 要求は作って見るまで分からない 10
4.5. 「モノ」を作る環境　生産ライン・自動化設備 10
4.6. 大規模制御システムの機能開発における連携不備 11
4.7. マツダの事例 11
4.8. industrie4.0 化を体系的に整理・可視化したツールボックスなど 12
5. システム思考とは 13
6. 纏め 15
7. 今後の課題 16
8. 参考 17
8.1. 参考文献リスト 17
8.2. システム思考ツールの例 17
8.3. 国内の取り組みの紹介 19
8.3.1. 横断型基幹科学技術研究団体連合（横幹連合） 19

8.3.2. 情報処理推進機構 IPA 20
8.3.3. 科学技術振興機構 JST 20
8.4. 企業の教育事例・情報系企業の場合 20
8.5. システムのライフサイクル対応とシステム・アプローチの関連性 21
8.5.1. トップダウン 21
8.5.2. ボトムアップ 21
8.5.3. アジャイル 21
8.5.4. リーン 21
8.5.5. 上記の複合 22
8.6. 『システムズデザイン・マネジメントとは何か』の紹介 22
編集後記 26
メンバー、委員会開催記録 27

はじめに

第四次産業革命とは一体何を意味しているのだろうか。加速・拡大する技術革新が社会に与える影響が大きくなってきている。今の時代は技術が大規模、高度化し、他の領域や他のシステムとの関係性が複雑になり、単独の領域の知識や経験に立脚する技術だけでは、課題を解決することが困難になりつつある。このような、時代の大きな潮流を、欧米ではチャンスととらえ、技術開発を進めている。日本はものづくりへのこだわりが強いためか、ものからシステムへの拡張ができずにいる。だから、システム思考が重要であり、モジュール化の考え方で整理し、システム化していかなければならない。

平成 29 年 11 月に開催された第 3 回 RRI 国際シンポジウムでは、ワールド・エコノミック・フォーラムの方より、「Think Systems not Technologies」という指摘を受けたが、この言葉に象徴されていると思う。

日本では、ようやく、データは取らなくてはいけないということを、少し意識し始めて、見える化が一歩前に進んだという状態である。デジタル化に対してデータは取るようになったが、それをどのように生かして行くかということは、まだほとんど着手できていない。AI の利用なども試行錯誤が始まったばかりである。

RRI／WG1 では、IoT 化が進み、設備機器がインターネットにつながる世界、機械・電気・情報通信系の様々なシステムシステムがつながる社会、System of Systems(SoS)の社会（特に次世代の製造業）を理解しようと、SoS 調査研究チームを立ち上げた。しかし、参加メンバーの議論から

- １つのシステムであるハード系の設計開発レベルでもうまく開発できていない課題がある。
- SysML を学んだが、設計開発レベルの課題を必ずしも解決できない。

という点が指摘され、何か根本的な点から違っているということが話題になり、それを議論した結果が本書となった。

システムを考える上で同様の課題に多くの方が直面していると考え、参考となれば幸いである。

1.　時代背景の再認識

我々は日々、ものづくりを取り巻く環境の様々な急激な変化を感じている。

- 製品や技術のコモディティ化が進み、新興国の製品・技術との差別化が難しくなってきている
- 製品の主な価値が存在する場所が、ハードウェアからソフトウェアに移行してきている
- すなわち、単独の企業または企業体内ですべてを賄う垂直統合型から，不得意な技術や不足する技術を他に求める水平分業へものづくりが変わってきている
- 一方で高い労務費を抱え、今後労働人口が減少する構造的な問題がある
- 更には、市場のニーズが、単なるモノの購入で満足するのではなく、エクスペリエンスを求めるコトに変わってきて、単純大量ではなく個性化・個別化し、さらに時間により次々に変化している。

これらの状況を踏まえて、ドイツではインダストリー4.0、米国ではインダストリーインターネットコンソーシアム、中国では中国製造 2025・インターネットプラスなど、欧米中では、ものづくりに関する新たな取り組みが国レベルで推進されている。

これらの結果、我々日本企業はこれまでにない新たな課題に直面している。

- Physical の世界を Cyber に取り込み、この Cyber と Physical を融合したものづくりへの対応
- モノからデータを取りだし、取り出したデータの利活用 AI の有効的な活用
- 人に頼った仕組み人海戦術ではなく、システム的対応
- Value Chain・Supply Chain・Engineering Chain・経営全体のコントロール
- つながる世界の相互信頼関係の構築
- 新たなエクスペリエンスを提供するコトづくり

これらを活用したり，これらに対応することにより、これまでの単独もしくは狭い領域の製品や仕組みが、多分野に跨って複雑に結びついて価値を生み出すことが可能となっている。これらは我々がこれまで経験してきたことの延長上にはなく，全く新しい取り組み・挑戦によりもたらされると言っても過言ではない。そのために、これまでにない新たな考え方・視点が我々には必要になってくる。

- モノ単独の視点ではなく、モノを取り巻く全てを俯瞰する視点
- 多分野にまたがる機能を統合する取り組み
- 目指すべきゴールを明確にし、それを実現する機能とアーキテクチャを構築する考え方
- ゴールの達成を評価する手法
- これら取り組みを効率的・効果的に進めていく考え方

この新たな考え方・視点・手法としてシステム思考・システムズ・エンジニアリングが注目されている。

2. 本書の狙い

2.1. 想定する対象の読者

本ガイドブックが想定している読者は、技術者をはじめとした、製品開発に関わる全ての人、さらには製造業ビジネスの経営者含めた関係者、を対象としている。

2.2. 読者に伝えたいこと

現在のモノ作りの傾向は、システムの複雑さが増しており、新しいシステムを実現することに伴う課題がこれまで以上に増えている。継続的かつ進化的に新しい技術が導入されるにつれ、要件は絶えず変化している。多くのシステムのライフサイクルが延長されている一方、個々の技術や特定の技術のライフサイクルは短くなっている。最近のシステムは、相互運用性要件とシステム・オブ・システムズ（SoS）の特性・内容を持ったものが、より多く見られるようになってきている。現在使用されているシステム（製品）の多くは、性能、信頼性、サポート性、品質、および全体的なコスト効率の面で、顧客／ユーザーのニーズを必ずしも満たしていないものが数多く存在する。

このようなものづくり環境の変化の中で、構築されるシステムは、変動性、不確実性、複雑性、曖昧性の特徴がますます顕著になってきている。その代表例が、IoT システムであろう。当初の想定とは違うものがつながり、複雑さが増す中で、その複雑さをコントロールする手段が必要になる。その手段が、システムズ・エンジニアリングの手法である。本ガイドブックで読者に伝えたいことは、システムズ・エンジニアリング手法の底流にあるシステム思考を理解いただく一歩として、以下に示す内容である。

- システム**思考**が重要
- システム思考で考えるとは、ユーザー環境下でのユーザーの価値と運用を考える事
- システムズ・エンジニアリングの手法（例えば SysML など）
  を適用しているからと言ってシステム思考・アプローチとは言えない

2.3. 本書の位置づけ

本ガイドラインは、日本のものづくりの限界を共有し、システムズ・エンジニアリングの推進力を備えた設計開発プログラムの実施のための考え方の視点（システム思考の基本）を提供し、システムズ・エンジニアリングの手法を適用してみようとする動機付けを行うものである。

- 従来型のある狭い領域の技術（機械系、電気系等）に携わり、携わっているサブシステムあるいはコンポーネントの入出力だけに注意を払い、ボトムアップ的にシステムを開発するものづくりの手法が限界にきている事実を共有し、気づいていない人に気づきを与え、‘問題意識’として、今の延長上に解はないという共通認識を醸成する。
- システムズ・エンジニアリングの観点から新しいものづくりのアプローチの示唆を提示し、あるべきシステムの姿とシステム思考を普及させるための初心者向け、導入初期のガイドブックとする。
- 尚、手法や方法論ではなく、考え方の気づきを与えることに重点を置く。

2.4. 対象とするシステム

- マルチドメイン(機械・電気・制御・情報通信など)を融合するシステム、例えば、自動車の ABS の例のように、機械系と電気系があってさらにソフトウェアが介在しているというような複合要因があるような製品
- （グローバルで）企業間連携を必要とするシステム
- 状態爆発を起こす可能性があるシステム
  - システムを構成する要素が膨大になり、外部要素との関係性が膨大になるシステム
  - システム全体のシステム特性（信頼性、可用性、保守性、安全性）の立証が必要なもの

3. システムに関する定義

システムなどシステムに関連する用語の定義は、多くの文献などに存在する。ここでは代表的な定義を取り上げ、用語の拠り所とする。

3.1. 既存のシステムの定義

定義された目的を達成するためのサブシステム、要素の組み合せ。ハードウェア、ソフトウェア、 ファームウェア、人、情報、技術、設備、サービスおよび他の支援要素を含む。(INCOSE)

構成要素の組み合わせからなるシステムを組み合わせたものもシステムであり、複数階層の構造（ビルディングブロック）を取る。

3.2. システム関連用語の整理（既存の定義）

3.2.1. システム科学

複数の科学分野（自然、生命、医療、社会科学など）をシステムとして把握・分析し、横断的に活用すること目的とする学際的な学問体系である。

3.2.2. システム・アプローチ

様々な専門分野に限定された問題解決の方法論ではなく、複数の分野（ex. 数学、物理学、社会学、工学、経営）の視点から物事をシステムとして俯瞰的に眺めて、単独分野でのアプローチでは解決できないような複雑な問題を解決する方法論。

3.2.3. システム指向

複雑なシステムに取り組む際に、設計者が問題の複雑さを容易に把握するためにシステム・アプローチを用いるシステム重視の考え方。

3.2.4. システム・セオリー

システム・セオリーは、個々の学問分野（自然、社会、科学など）の相互依存関係をシステムとしてとらえ分析する学際的な理論。

3.2.5. システムズ・エンジニアリング

システムズ・エンジニアリングは、効果的なシステムの実現を可能にするための学際的(インターデシプリン)な開発アプローチである。開発サイクルの早い段階で顧客ニーズと要求機能を定義し、要件を文書化した後、運用コストとスケジュール、パフォーマンス、トレーニングとサポート、テスト、廃棄、製造といった全ての問題を考慮しながらシステム設計、構築、検証を進めることに重点を置く。全ての専門家と専門グループをコンセプトから生産に至る構造化された開発プロセスにチームとして組み入れる。

ユーザーニーズを満たす高品質の製品を提供することを目標として、全ての顧客のビジネスニーズと技術ニーズを考慮する。(INCOSE)

3.2.6. システム・オブ・システムズ

複数の構成システムが相互に複雑に作用し、個々の構成システムの振る舞いからは予測できないような創発的な振る舞いを生じさせるシステムであり、下記の特徴(*1)を持つ。

- 運用の独立性：構成システムは個別に動作し、独立に運用
- 管理の独立性：構成システムの管理権限は、各構成システムで独立に保有
- 進化的開発：機能や目的が追加、削除されるなど進化的
- 創発的振舞い：構成システム単独では実現できない目的を、SoSとして実現
- 地理的な分散：構成システムが離れており、構成システム間で情報を交換

*1：“Architecting Principles for Systems-of-Systems”, Mark W. Maier (1998)

3.2.7. システム・デザイン

システムの目的、目標を達成するために、アーキテクチャ 、機能、構成モジュール、インタフェース、 データのみならず、社会や周りへの環境影響などありとあらゆる関係を考慮してシステムをデザインすること。

4. 身近な事例

以下に示す事例は、システム思考に関連する失敗事例、成功事例を紹介する。又、手法に関する海外の取組み事例なども含めた。

4.1. 実装優先開発

機能の実現のために思いついた手段の実装を優先するあまり、たとえ開発の中で想定外の問題が起こり本来満足しなければならない性能が徐々におろそかになっていっても、最後までその手段の実装に執着してしまった。システムの目的、狙いを見失って実装が先行してしまう。

4.2. 暗黙知開発

システムとしての要件を明確にする前に、各コンポーネントの図面を書くことから開発がスタートし、コンポーネントの試作が出来、システムが組上がり、そこから初めてシステム要件の検討がスタートする。そこには開発経験豊富なエキスパートが集まり、彼らの暗黙知をベースに、あれはどうだ、これは見たかと開発の主導権を握る。開発の終わりは彼らが黙れば終わり。いつも開発の最初に目指す姿は見えておらず、開発の最後に元々目指していたことも見当たらない。これは経験が問題を潰しこむやり方であり、経験に無い問題は発生してから潰していくしかない。結局、開発のエビデンスが残らず、要件から結果までのトレーサビリティが取れない。そのため品質確保（エキスパートを黙らせる）のためには、念には念を入れた人海戦術に頼らざるを得ない。

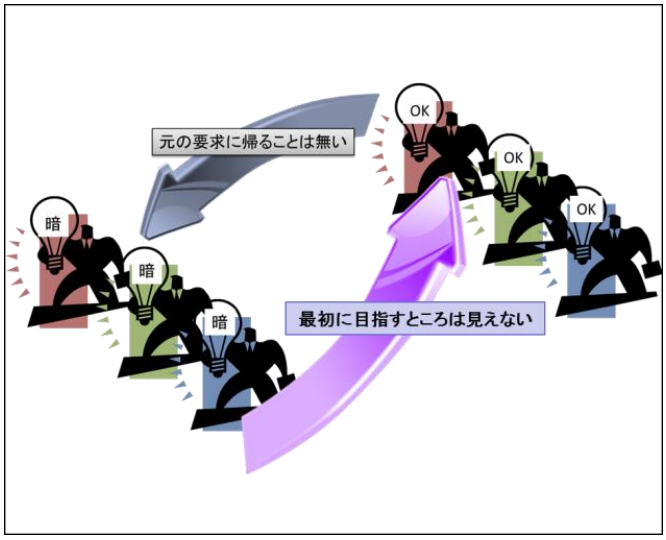


図 4.1　暗黙知開発

4.3. 変化を好まない変化への対応

既存の製品の取り付ける環境や、使用方法の変化に対して、とにかくまずは既存の製品の改良で対応する。しかし実際の環境や使用方法で評価をすると、変化をとらえきれておらず次々と不具合を経験することになる。システムの取り巻く環境の変化によるそもそもの要求の変化の事前検討が出来ていない

事例：

車両の室内搭載の電子制御ユニットを、エンジンルームに搭載を変更した際に、そのままエンジンルームのアクチュエータに直付けした。そのため電気系インタフェースのための接合面が多く残り、更に電子制御ユニットが横置きだったため、水の侵入経路が多く、水が一旦入ってしまうと抜けにくい構造になっていた。結果的に水の侵入に悩まされ、特に製品の運搬や保存時の水の侵入については検討が不足していた。

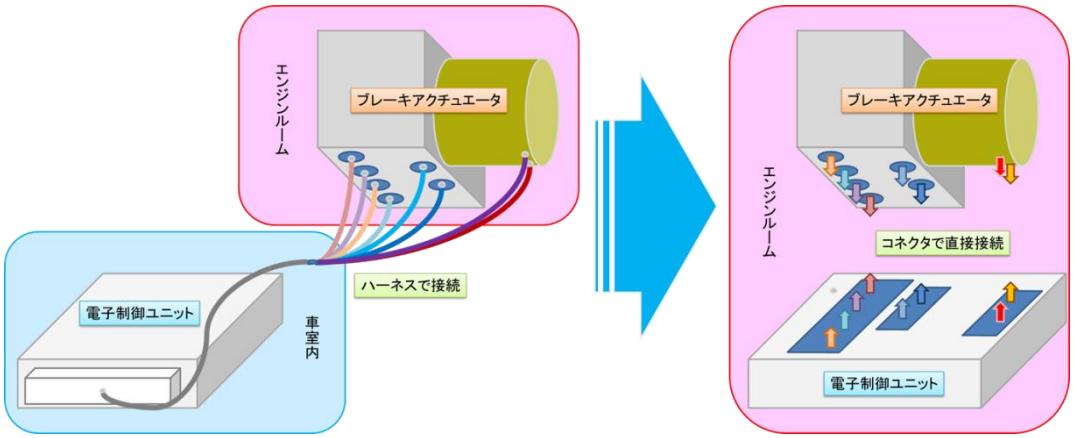


図 4.2　自動車の電子制御ユニット

4.4. 要求は作って見るまで分からない

顧客に製品を持って行き実際に動かして、その評価結果をもらいながら要求を満足させていく。製品の対象範囲や、ライフサイクル毎の環境全てを網羅し全ての要求を満足するまでに、結果的に非常に長い時間を必要とする。

事例：

車両制御の場合、製品が車両に搭載されてからのフィードバックがかかることで要求の見直しされる。このために、車両制御が熟成され、変更がほとんど必要なくなる状態になるのに 10 年位かかることがある。

4.5. 「モノ」を作る環境　生産ライン・自動化設備

新製品の生産ライン・自動化設備の立ち上げでは、製品自体の製造品質確保と投資対効果が極めて重要な KPI となる。特に短納期での立ち上げが必要な場合、生産ライン・自動化設備自体の運用や製品ライフサイクルを視野に入れた機能設計が疎かになりやすい。自動化設備の例では、省人化○○名・マシンサイクルタイム△△以内を守ることが目的化し、立ち上げ後の製品仕様変更や突発障害に対する対応に対する要件の検討が不十分となりがちである。結果として、安定稼動までのリードタイムが長期化したり、新製品への更なる対応において多額の投資が必要となったりするケースが多々あった。「モノ」を作る環境においても、その環境に携わる様々なステークホルダの整理、中長期視点でのあるべき姿のデ

ザインが極めて重要である。これらをデザインしてロードマップ化しつつ、前述した KPI を達成するためのベストエフォートを選択し、戦略的にあるべき姿へと導く取り組みが必要である。

富士通では、*QCD* の向上、市場先行投入、製品ライフサイクルへの柔軟な対応、ライン・設備運用の効率化等の様々な狙いを整理し、*ICT* を活用した仮想検証をベースとしたコンカレント生産準備を企画。*3D-CAD*、ロボットシミュレータ、*PLC* シミュレータ等をミドルウェアで接続し、設備全体の動作を忠実に再現可能な設備制御プログラムの開発環境を構築。電気・伝熱・光学等のアナログ要素が多数存在する分野の設備開発においては、各要素の *1D* シミュレーションモデルを統合したシミュレーション環境をプログラム開発環境と統合し、製造品質を実機レスで検証可能な環境としている。開発した設備制御プログラムは、実機のロボットコントローラ、*PLC* に適用可能であり、工場に赴かなくても障害の再現～恒久対策の検討・改修を迅速に行えるようにした。また、一部の事例では、設備開発期間を *6* ヶ月から *2* ヶ月に短縮、量産立ち上げ後の機能追加や製品仕様変更に伴う設備改修期間を *5* 日から *1* 日に短縮することができている。

4.6. 大規模制御システムの機能開発における連携不備

比較的大規模な生産ラインにおける制御システムは、数十台の PLC(Programmable Logic Controller、シーケンス言語，高速処理)による Level 1 機能群と、数台の計算機（Windows 等、C 言語記述、比較的低速）による Level 2 機能群が、高速 LAN で接続されている。

ある新機能を導入するための仕様を作成したが、Level 2 と Level 1 それぞれの機能群に改修を加える必要があった。しかしながら、それぞれの担当者の連携が不足し、各自の勝手な思い込みにより仕様およびソフトウェアを設計し、社内試験、現地試験も十分行わなかった。その新機能を顧客の前でオンラインするまで、不具合があきらかにはならなかった。木（要素）だけを見て森（システム）を見なかったために起こった不具合である。

4.7. マツダの事例

システムズ・エンジニアリングは初心者に取り組みにくい印象があるが、よく似たトップダウン型の概念を持つ「機能エンジニアリング」という実践的な手法が提案されている[1]。具体的なエンジン開発に適用した例があるので紹介する。

マツダ社の事例：

マツダのスカイアクティブエンジンは、機能エンジニアリングというトップダウン手法で開発された。会社からの「30%の燃費向上」という要求に、エンジンのそもそも論から考え直し、「ガソリンエンジンで世界最高の熱効率 40%」という高い機能目標を開発リーダーが提示したことが成功の起点となった。性能ではなく高い機能目標(ステートメント)をメンバーに提示することこそが、開発プロセスの出発点であり、開発リーダーの仕事である。

機能エンジニアリングは次の 5 つの段階からなる。

＜Step1＞ステートメントの設定：本来的に求められる機能である「源機能」を見極めた上で、機能特性と機能水準を明確にする。(例：熱効率を世界一に)

＜Step2＞目標機能水準の設定：高い機能目標であるステートメントをもとに機能を物理特性で表現し、目標となる具体的な数値を決める。(例：熱効率を 40%に)

＜Step3＞カラクリ展開：機能に関わる物理的な各機能の因果関係を視覚化して各因子にカラクリ展開し、CAE モデルも活用しながら具体的な物理式に置き換えて定量化する。(例：熱効率→(第 1 展開)燃焼特性・冷却損失・機械抵抗損失・排気損失・ポンプ損失→(第 2 展開)・・)

＜Step4＞サクセスシナリオの構築：カラクリ展開を分析して解明し、各因子に具体的な水準を割り付け、目標水準を達成するためのサクセスシナリオを十分に時間をかけて作る。

＜Step5＞製品開発のための構想設計：サクセスシナリオに従って各因子の物理特性水準を達成するため、具体的な手段を決めるための構想設計を行う。この段階で初めて図面に落とし込む。これにより機能が獲得され、自ずと品質も保証される。

[1]「つくりたいんは世界一のエンジンじゃろうが！」(羽山 信宏 著、日刊工業新聞社 2014) 。

4.8. industrie4.0 化を体系的に整理・可視化したツールボックスなど

独 VDMA ドイツ機械工業連盟は 2016 年、SME 向けに Guideline Industrie 4.0 を発表。テクノロジーが導く industrie4.0 化を体系的に整理・可視化したツールボックスとワークショップ形式によるビジネスモデルクリエーションを行うフレームワークが示された。ツールボックスは industrie4.0 のコンセプトに従って製品・生産をどのように変化・進化させていくかを示したものであり、これを用いて現状の分析やアイデア創出を行う。アプリケーションカテゴリと industrie4.0 レベルで分類・細分化されており、段階的・戦略的な industrie4.0 化を促している。実際の開発においては、デザイン思考のアプローチが採用されており、ビジネスモデルの構築にフォーカスしている。製造業では得意とはいいがたいビジネスモデル構築であるが、前述したツールボックスを用いて分析、創造、評価をすすめることで、実践的かつ有益なワーキングを可能としている。こういったガイドラインがトップダウンアプローチで示されている点は、わが国としても参考にすべきである。

5. システム思考とは

ここに挙げた事例だけからも、システムに対する見方はいろいろあることが見えてくる。その中で、最初に意識すべき事が、システム思考である。

失敗の事例の多くは、対象とする製品を中心に、その製品の入力、出力の要求を分析するために、事前の検討が、製品を取り巻く様々な環境、事象全てに対して、十分分析が行われていない。その結果、分析の漏れ抜けが不具合を生み、大きな手戻りを発生させて来ている。

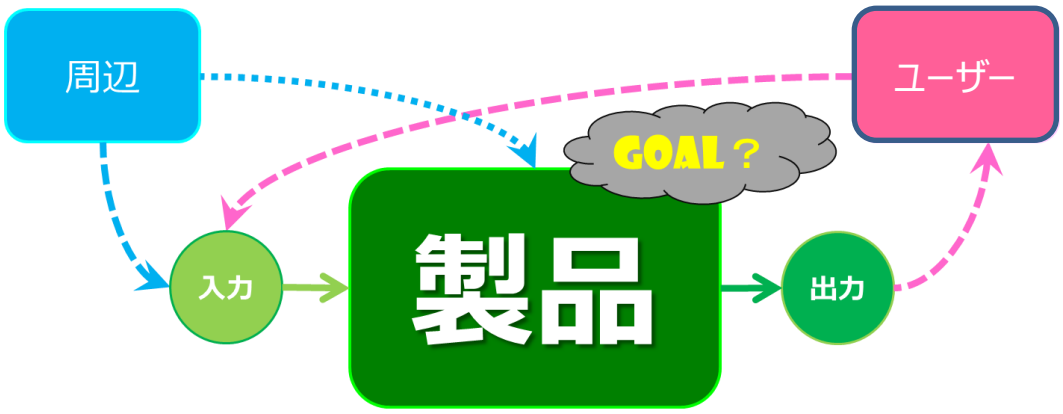


図 5.1　失敗事例の構造

基本的には製品そのものだけで、製品そのものの良し悪しを決めることはできず、それを使用するユーザーによってのみ、その製品の良し悪しを最終的に決定できる。また、ユーザーとその製品の周辺には、それらを取り巻く環境があり、更には、ユーザーがその製品を使用していくライフサイクルがある。製品は、これら全てとの相関関係があり、その相互作用から要求される振る舞いが機能となりシステムアーキテクチャを決定しその製品への要求となっていく。これらユーザーと製品とその周辺環境との相互作用を、如何に空間的・時間的に俯瞰し、事前にその製品への要求を網羅的・階層的に導きだしていくための方法論として期待されるのが、システムズ・エンジニアリングである。

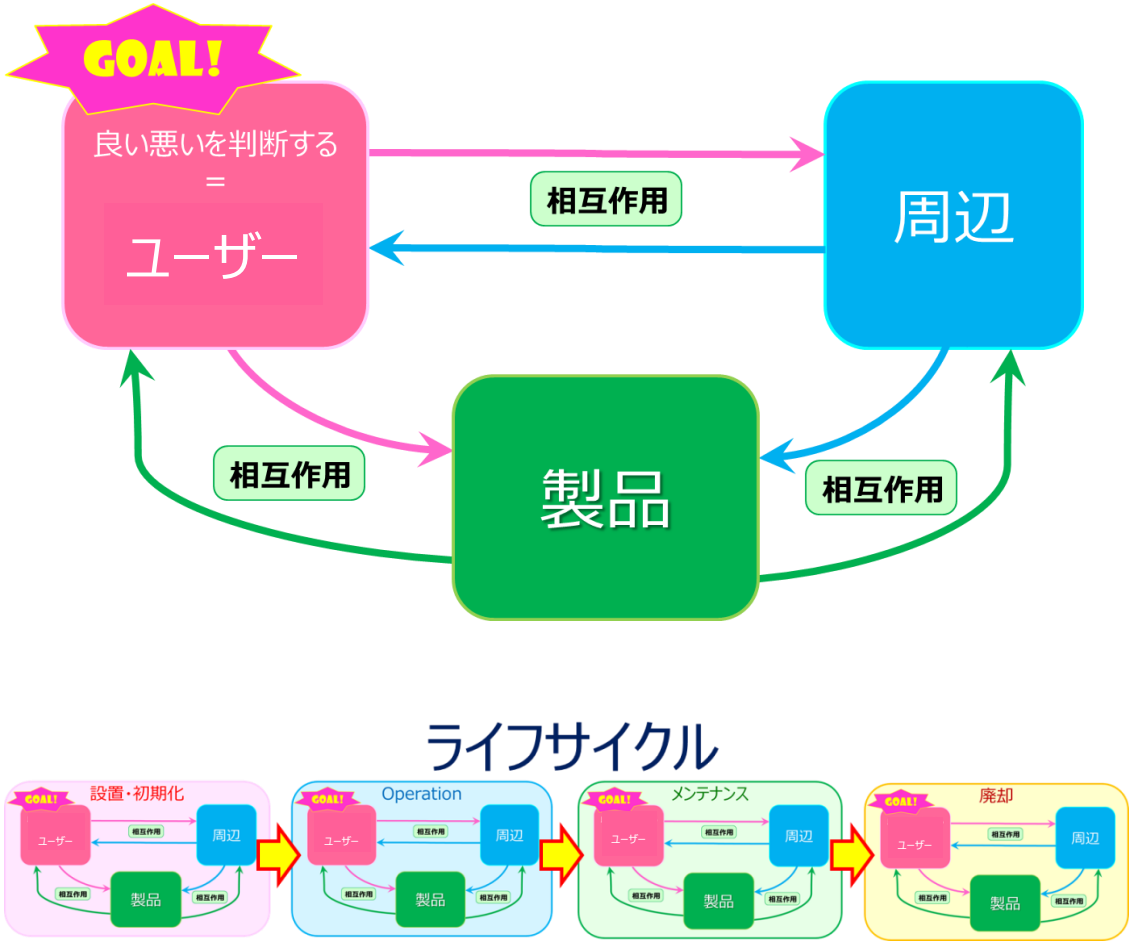


図 5.2　システム思考の構造

このようにユーザーの目指す Goal（Why）を明確にし、この Goal を達成するために、ユーザー、周辺、製品の相互作用からどんな振る舞い（What）が必要なのか、そしてこれら振る舞いを如何に実装（How）していくのかを、自然言語による抽象レベルから、実際の実装までを、丁寧に、Why⇒What⇒How を紐付けながら繰り返しによって落とし込んでいく。すなわち、各プロセスの仕事の成果を、Goal とのトレーサビリティが取れ、各プロセスの中で、Why⇒What⇒How の結びつきを見ながら業務が遂行できるようになる。これによって、Goal を見失わず最適な製品が、手戻り無く作り出せるようになれる。

6. 纏め

本報告書では、もの作りにおいてシステムズ・エンジニアリングプロセスを実践していくために上流プロセスで最も重要だと思われるシステムアーテクチャの決定の際に、システム思考を駆使して取り組むことの必要性に焦点を絞って説明している。

システムアーキテクチャを決定する際に、機能は一つでも物理解は複数存在するケースがある。その中から最適なものを選択する際に、システム思考による全体俯瞰が必要になる。システムを企画し・開発・運用していくには、製品・技術の視点ではなく、まずは、視点を拡げて、ユーザーのゴール、ユーザー・周辺の環境・製品の相互作用から考える『システム思考』が重要である。

しかし、事例などを見ていくと、その上で、取り組むべきことがまだいろいろあり、その例を以下に示しておく。

- システムズ・エンジニアリングなどの手法を使い、場合によっては情報技術を駆使したシステムを導入し管理することが必要になる。
- 組織面でも体制を整え、マネジメントを工夫する。

など、この部分についてまでは、今回記載していないことを最後に触れておく。

システム思考を磨いて、システムとして取り組んでいくためには、組織やマネジメントやツール化等、まだまだ、取り組まなければならない課題がある。このような事を世界では、体系立てて実施している。それを、追従していくために、まず、第一歩を踏み出すことが重要である。

7. 今後の課題

今回の調査活動においては、最初のアプローチとして、システム思考に着目し、要求仕様を全体俯瞰により、相互作用をしっかり考えて、固めるという考え方を中心に調査してきた。しかし、それが実行可能になった後、大規模システムを開発するにためには、方法論であったり、組織論、マネージメントレベル等を、きちんと考え、実行に移す段階は実現できていない。まず、発想の段階を根付かせるという意味では、良いのだが、次の段階として、プロジェクトにして、遂行していくための方法論が必要であり、マネージャの役割だとか、安全性の立証など、もともと、それを実行に移すための、組織構造、ツール、手法がなければ難しい。こういう事を、どのようなスケジュール感で実施していくべきかを議論していく必要がある。

今後様々な関係者と調査や検討が必要な項目を列挙する。次年度以降の活動に期待する。

- 情報システム側でのシステムの捉え方と機械・電気のハードウェア側から見たシステム捉え方の共通点と相違点の明確化
- システムズ・エンジニアリングの全体像の把握
- 日本でのシステム思考、システムズ・エンジニアリングが未普及の原因
- 日本のボトムアップ文化に合ったシステム思考の検討
- システムを企画、開発、運用する上での責任の連鎖を担保する組織論やマネジメントの必要性と体系化
- 機械、電気製品から始まり、部品点数の観点（自動車→飛行機→大型客船など）、複雑化の観点（自動運転や航空システム、スマートエネルギー、スマートシティなど）それぞれの段階にあったアプローチ
- 分野別事例の整理やツールの体系化
- ガイドブックの検討（今回が入門編とすると、導入編）
- 海外を含む他の活動との情報交換

8. 参考

8.1. 参考文献リスト

(1)『**システムデザイン・マネジメントとは何か**』：慶應義塾大学大学院システムデザイン・マネジメント研究科編，慶応義塾大学出版会
今なぜシステム思考が必要か，システムとは何かなどの基本的なことが，整理された平易な文で書かれた入門書．8.6.参照。

(2)『**つくりたいんは世界一のエンジンじゃろうが！**』：羽山信宏，B&T ブックス 日刊工業新聞社
非常に面白いですね．システム思考で何でもやらなければならないと突き詰めると，たぶん（私は）窒息死するので，その一歩手前で有効な解の一つと思います．

(3)『**一般システム思考入門**』：G. M. ワインバーグ、紀伊國屋書店
本書は、発行当時米国の高校、大学、企業で広く扱われ、高校の数学の知識があれば読めるとされている一般システム理論の入門書。

(4)『**デザイン・ルール**』：C. Y. ボールドウィン他、東洋経済
システムを構成するもの、その構造化＝モジュール化の研究の第一人者による。モジュールかとは何かを詳細に記述し、それにより産業が進化していくプロセスを理論化。ビジネスを動かす「推進力」の本質を解き明かす。

(5)『**バーチャル・エンジニアリング**』：内田孝尚、日刊工業新聞
世界的に進む「設計」と「ものづくり」融合。本書は、そのキーテクノロジであるバーチャルエンジニアリング(VE)について、我が国製造業での導入課題と見通しを明らかにする。積極的な欧州の動きを報告しながら、VE の狙い、対応の遅れが何をもたらすのかを解説し、警鐘を鳴らす。

8.2. システム思考ツールの例

日本の製造業は、定められた“要求仕様”に沿って誠心誠意ものづくりをすることを得意としている。それは、生産技術に強みがあるからである。その強みに安住し、以下のような状態になっているのではないかと思われる。

- 製品自体ではなく、「製品を介して実現されるユーザー価値」に注意を払っているか？「使い方は顧客の問題だ」としていないか？
- 一方、顧客自身が選択する仕様がユーザー価値に最適なものだとどうやって判断できるのか？その判断をも顧客の責任として切り分けていないか？（バリエーションが一定数を超えてしまうと、顧客にも選べなくなる。本来的には、製品について一番詳しい人こそが、顧客の状況を聞いてそれに最も合っているモノを推奨することができるはずである。）
- 「製品ユーザーの“本当の使い方”」の把握が出来ているか？ユーザーは千差万別だが、その全て（千差万別の“仕様”ではなく、千差万別のユーザーの“使い方”）に対応できているか？

生産技術単体では、「設計開発」や「ソリューション」に比べ付加価値を大きく生み出すことが難しい状況になっている。いい「もの」を作るだけでは生き残れない時代に入り、常にユーザー価値を中心としたバリ

ューチェーンとすることが求められている。ものづくり企業が持つべきは、「ユーザーの“使用”にコミットした提案力である。それを支援するのが、製品のアーキテクチャ構築段階における顧客価値機能から物理構成解案に落とし込む AI コンフィグレータ（tacton）である。

これまでは、製造業のビジネスに内在する課題として、以下に示すように、顧客が本当に望む「使用」を実現する最適な製品を提供できている保証はなかった。

- 大型電動機や製造プラントの電設のような商材を扱う受注開発型ビジネスでは、顧客からの要求（仕様）に落とし、自社保有構成部品あるいはカスタマイズ可能範囲のオプションとして細かく区切り、1 つ 1 つ順を追って設定していくのが大まかな流れである。
- このとき、要求の背後にある真の理由、使い方にアプローチすることは極めて困難である。理論的にはより最適（顧客にとって：効率的にほしい機能を取り出せる（使用できる）組み合わせを獲得できること、自社にとって：自社製品の価値をより高く認識してもらえること）に近いものを突き詰める余地を残したまま、製品が現実世界に実装されることになる。

仕様決定は、性能・機能訴求型から顧客価値機能訴求型へ転換していく必要がある。顧客との会話は、「要求仕様」ではなく「使用目的・使用環境・購入価値基準」等で交わし、結果を実装していく。顧客が望む「こと」に最適な「もの」にするには、自社の製品モジュールを正しく知る必要がある。過去の実績は最適ではない。選ばれるべきは最適な組み合わせを得ることである。

- 実績は、「要求仕様」を決めることによって出来上がったものであるから、改めて使い方をパラメータとしてプロットすると、必要とされないラインナップを提供しているケースもある・
- AI コンフィグレータの手法によって外れ値を棄却することにより、製造都合ではなく顧客ニーズに対応したモジュール化と、予め備えておくべきモジュールの絞り込みが可能となる。
- 「モジュール化」とは、組みお合わせによってバリエーションの多さをカバーするものであり、最適な組み合わせは、使用と仕様の関係がわかっていて初めて可能となる。

AI コンフィグレータにおける構成情報の扱い方は、以下のとおりである。

① システム全体を通した構成ルール（組成＋要素の属性間ルール）を入力する、「レイヤー間＋レイヤー内＋全体共通の関係」
② 構成要素毎の可能な部品リスト（部品ごとの属性を含む）を紐付ける
③ 上記①と②から、可能な構成組み合わせ（1 システム分）を全て計算（構成ルールと属性同士の適合関係で、100 億通りでも可）
④ 条件パラメータ、最適化基準を与えて、可能な構成組み合わせ全ての中から一つを選択する。

AI コンフィグレータのような支援ツールの助けを借りながら、システムズ・エンジニアリングの手法を適用し、顧客価値に寄り添ったもの作りを実現していけば、以下のような変革の推進力がアップし、グローバルものづくり強者への仲間入りが可能となる。

① 顧客価値に寄り添った選択肢を提示するだけでなく、期待以上の価値をパッケージ化したソリ

ューションとして見積提案することが可能になる。

② 製造現場が作りやすい「もの」を受注することによるコスト削減や、営業が高く売れる「もの」を作ることによる売り上げ拡大をも実現できる。

③ 世界中の様々な価値観を持つ顧客・市場をフィールドとして、グローバルレベルの競合と伍して戦えるものづくり戦略を立てる事が可能になる。

http://www.tacton.com/ja/

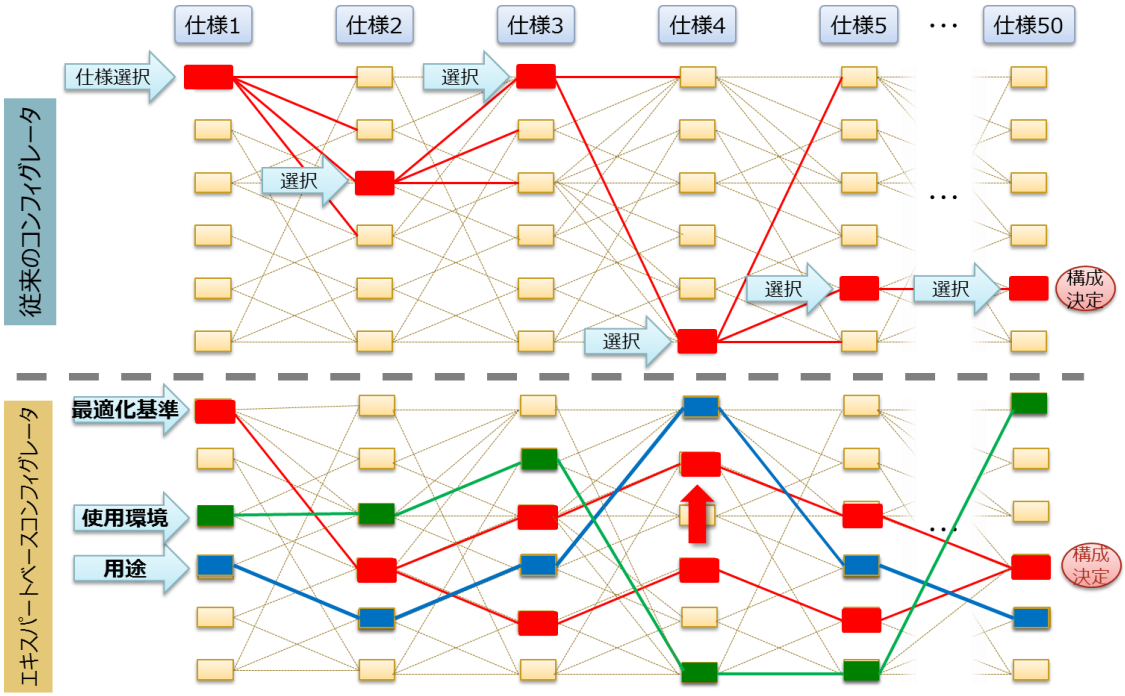


図 8.1 従来のコンフィグレータとエキスパートベースコンフィグレータ

8.3. 国内の取り組みの紹介

8.3.1. 横断型基幹科学技術研究団体連合（横幹連合）

横幹連合は、文理にまたがる 43（設立時）の学会が、自然科学とならぶ技術の基礎である「基幹科学」の発展と振興をめざして大同団結したもので、限りなくタテに細分化されつつある科学技術の現実の姿に対して、「横」の軸の重要性を訴えそれを強化するためのさまざまな活動を行う。この中の活動として、経済産業省の製造技術実態調査に協力し，木村英紀顧問を主査とし、産官学からの参加者を得て，「第 4 次産業革命とシステム化研究会」が開催されを務められた（2016 年 9 月～2017 年 3 月）．成果は平成 2 8 年度「製造基盤技術実態等調査 （第 4 次産業革命における「知」のシステム化対応の実態調査） 報告書」(平成 29 年 3 月 31 日発行)として纏められ、第 4 次産業革命とは，ICT の進展がもたらしたシステム化の新たな波であるとして， 今日の日本の課題を掘り下げるとともに，人材育成や研究取組み体制などの施策が提言。

http://www.trafst.jp/IRsys.html

又、2017 年 11 月 7 日には、「システム・イノベーション」シンポジウムが開催され、製造業、サービス業、科学技術、経営など多くの領域で我が国の国際競争力の長期低落傾向はとどまるきざしが見えない状況だが、この根本的な原因の一つとして、現代におけるイノベーションの源泉であるシステム化への認識が遅れ、システム化の技術が未成熟でそれを担う技術者・研究者が絶対的に不足している状態にあることを指摘。（尚、ここで使われているシステム化とは、システム・アプローチや一般システム理論（システムをメタに捉えた学術的捉え方）やいわゆるシステム科学全般を指す広い概念を指していると思われる。）
http://www.trafst.jp/IR_sympo_report.html

8.3.2. 情報処理推進機構 IPA

近年、第 4 次産業革命をもたらすと言われている IoT の進展や、独立したシステムが互いに関係し合って価値を提供する SoS（System of Systems）のような複雑なシステムの増加、また、システムに対する要求の多様化、高度化、複雑化、予測できないリスク、関連する分野の調和や統合の欠如など、システム開発を取り巻く課題も難しくなってきており、解決が困難になってきている。

IPA/SEC は、このようなシステム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、欧米では一般的な製品やシステムへの適用が進み、有効性が認知されているシステムズ・エンジニアリングが日本においても有効と考え、産業界に普及・展開する活動に取り組んでいる。研究会や 「システムズ・エンジニアリング導入の薦め」や海外動向の調査報告などが発行されている。
https://www.ipa.go.jp/sec/our_activities/se.html
https://www.ipa.go.jp/sec/secjournal/048.html

8.3.3. 科学技術振興機構 JST

研究開発の俯瞰報告書　システム・情報科学技術分野などが発行されている。
https://www.jst.go.jp/crds/report/report02/CRDS-FY2016-FR-04.html
https://www.jst.go.jp/crds/report/report02/CRDS-FY2015-FR-06.html
https://www.jst.go.jp/crds/report/report05/CRDS-FY2014-WR-14.html
https://www.jst.go.jp/crds/report/report05/CRDS-FY2014-WR-08.html

8.4. 企業の教育事例・情報系企業の場合

富士通では、部門/職種・専門教育として各領域において必要な専門スキルの開発を行っている。近年、若手からミドルクラスを対象に、「ハイポテンシャルな人材を選抜」し、「実践を通じたアクションラーニング」と「修羅場経験（チャレンジングな配置）」を通じて成長の場を与えることを柱としている。特に、イノベーション領域でのビジネス拡大に向け、多様な人材とコラボレートしてビジネスを創造できる組織・人材の開発を強化。社内の様々な部門のエンジニアが集結・チームを形成して、デザイン思考・システム思考を繰り返し実践。ユーザーと一緒にビジネスを創出し続けるビジネスリーダーとなるべく、ファシリテーション等

のベーススキルの育成や他企業との合同研修にも積極的に取り組んでいる。

8.5. システムのライフサイクル対応とシステム・アプローチの関連性

システム思考が行えた前提で、大規模なシステムや複数のシステムが協調して動作するようなさらに複雑なシステムを開発するために、開発アプローチ（方法）を選択することは重要である。開発アプローチには、主に下記の方法がある。

8.5.1. トップダウン

ウオーターフォール型とも呼ばれる古くからある開発アプローチである。
開発工程を区切り、初めに企画、要求定義、設計仕様、実装仕様、テスト仕様などの詳細を文章に起こしてから、開発スケジュールに沿って上から下に向かって開発を進める。基本的には開発途中での仕様変更が無くトップからのマネジメントが容易な一方、全ての要求事項を事前に把握しきれず、開発途中での仕様変更によるスケジュールの遅延や追加費用の発生などが起こりがちである。

8.5.2. ボトムアップ

下位の個別モジュールを実装し、それを組み上げて全体のシステムを開発してゆくアプローチである。早くから下位モジュールの実装、検証ができ、現場の意見が反映されやすい特徴がある。一方、組み上げ時にモジュール間の動作不整合や部分最適が判明し、開発の手戻りが生じるなどのデメリットがある。また、自己保有の技術に制限され、新たな技術の導入がされにくい場合がある。

8.5.3. アジャイル

アジャイルとは、「素早い」という意味で、顧客とエンジニアが協力して要件を定義し、1～2 週間で開発できる単位の設計、実装、テストの繰り返しにより完成度を高めていく開発アプローチである。要件は途中で変更、追加される前提で繰り返しテストされるため変化に強いロバストなシステムになる利点がある。一方、開発工期や、システムの機能変更など管理がしにくい場合がある。

8.5.4. リーン

米国における日本の生産方式の研究から生まれた開発アプローチであり、無駄の排除を中心として 7 つの原則を定めている。

原則１　無駄をなくす：開発過程における無駄を徹底的に排除する。（未完成のコード、仕事の掛け持ちのオーバーヘッド、不要な機能の実装、作業の引継ぎなど）
原則２　品質を作りこむ：誤実装はできるだけ早い段階で無くす（顧客参加により要求仕様理解の齟齬を無くす、テスト駆動開発により常にテストプログラムが通るように実装を進めるなど）
原則３　知識を作り出す：開発時の失敗をナレッジとして蓄積し同様な問題を予測、防止する。
原則４　決定を遅らせる：必要のない段階で、不十分な情報に基づき物事を決定することは避け、必

要な時期に状況が見えた段階でより精度の高い決定をおこなう。

原則５　早く提供する：顧客に早期に動作を見せて、要件の齟齬を確認し、場合によっては追加の要件を出してもらい早期に要件の修正をおこなう。

原則６　人を尊重する：開発メンバーが自律的に動き、協調することにより、個々の創造性以上の成果をあげる。

原則７　全体を最適化する：モジュールの部分最適を避け、常に顧客にとっての価値を最大化することを意識して全体最適化を目指す。

8.5.5. 上記の複合

開発手法はウオーターフォール、アジャイル、リーンなどそれぞれ特徴があり、開発規模、対象などに対して向き不向きがある。それぞれの特徴を理解したうえで、有効な開発手法を選択することが重要であり、場合によっては、各種開発手法を複合的に適用することも考慮する必要がある。

また、システムの処理のスループット、CPU の性能、通信速度から製造装置の耐用年数、商品の流通等ありとあらゆる物理制約が存在するため、物理的制約に合致した機能の割り振りをおこなうことが重要である。

8.6. 『システムズデザイン・マネジメントとは何か』の紹介

文献(1)は、システム関連の用語のみならず、システムの必要性、時代背景、多くの事例が掲載されており、システムへの理解を深めるための入門書としては適切である。その中の記述から、システムに関するキーワードを以下に挙げる。

① システムとは、複数の要素が相互に作用するとき、その全体をシステムという。システムは創発する。創発とは、要素の振る舞いを見ていても生じない振る舞いが、要素の相互作用するシステムにおいて見られることをいう。システムの例として、弓矢、矢そのもの、情報システム、交通システム、組織、社会、人間、言語、価値、幸福、感動、誠実、憎悪などが挙げられている。しかし、本ガイドラインで対象とするシステムは、これでは広すぎるので、以下の下線部で限定された定義が適切である。システムとは、定義された目的を成し遂げるための、相互に作用する要素を組み合わせたものであ、ハードウェア、ソフトウェア、ファームウェア、人、情報、技術、設備、サービスおよび他の支援要素を含む。これは、INCOSE Systems Engineering Handbook (Wiley)で定義されている。

② システムズ・エンジニアリングとは、INCOSE による定義として、「システムを成功裏に実現させるための、複数のディシプリン（分野）にまたがるアプローチおよび手段」とある。世界最初のシステムズ・エンジニアリングの標準は、1969 年に米国が空軍向けに制定した軍用規格であると言われている。同時期に行われた有人宇宙飛行計画「アポロ」はシステムズ・エンジニアリングによって成功したと言われる。

③ デザインとは、新たに何らかのシステムを創造し、そのアーキテクチャを定義し、その全体から部分までを適切につくりあげるという営みをいう。アーキテクチャとは、システムと外界との関係、または、システムを構成する要素とその要素間の関係をいう。

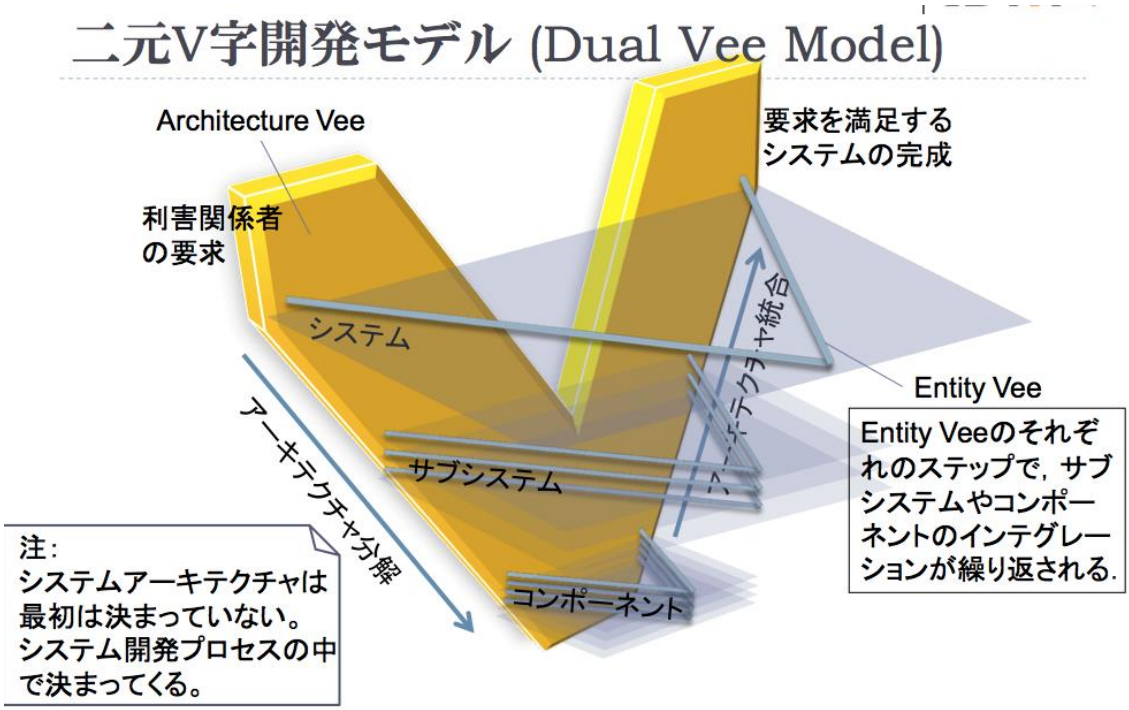
④ システム思考とは、ものごとをシステム（要素間の関係性）としてとらえることである。デザイン思考とは、チームで観察(Observation)、発想(Ideation)、試作(Prototyping)、を何度も繰り返しながら協創するイノベーティブな活動をいう。

⑤ システム X デザイン思考とは、「木を見て森も見る」というシステム思考の視点と，感性も駆使した視点で顧客価値を重視ながら新たな製品やサービスを見つけ出すような、いわゆるデザイン思考の視点の両方を持ちながら、デザインの対象に接していくことを指す。

⑥ モデルとは、計算や予測を支援するための、システムまたはプロセスの簡略化した記述。より広くは、物理世界にある実体、たとえば、システム、現象、プロセスを抽象化して表現したものである。

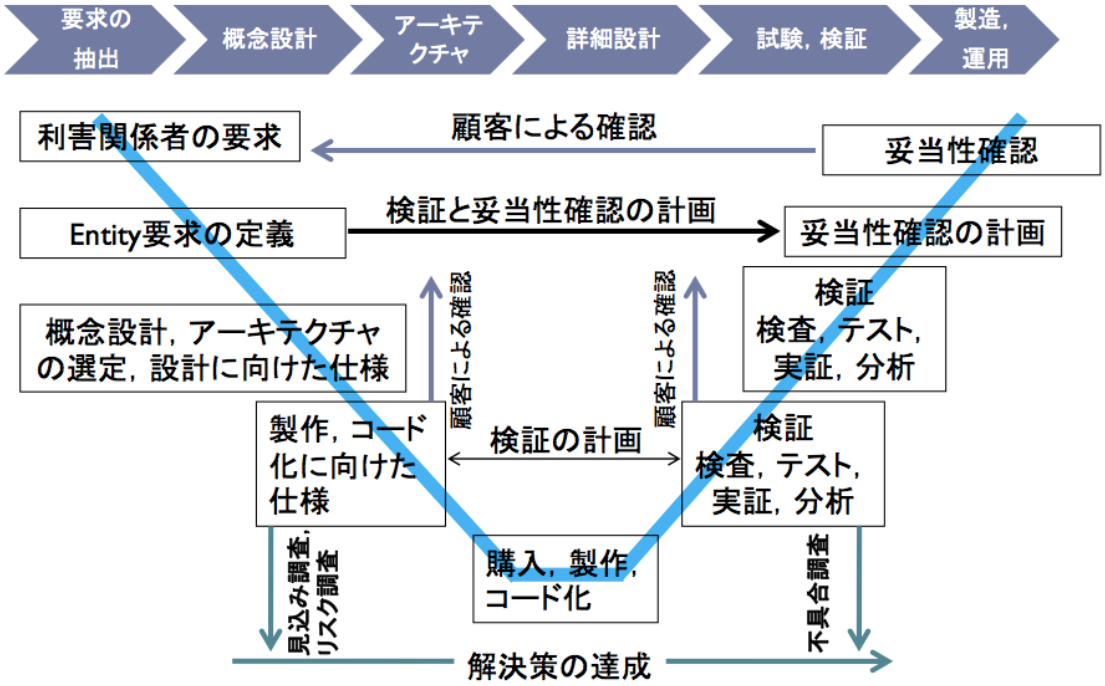
⑦ MBSE (Model Based Systems Engineering)として、INCOSE は以下のように定義している。コンセプト設計に始まり、開発以降のシステムライフサイクルにわたってつづく、システム要求、設計、分析、検証、妥当性確認に関するアクティビティを支援するためにモデリングを形式化して適用すること。従来、文書に基づいて進められたシステムズ・エンジニアリングは管理、統制が難しかったが、情報を持ったモデルにより形式化できてきた。

⑧ ２元 V 次モデル(Dual Vee Model)は、より大規模なシステムを扱うのに適していると考えられる。



参照：

https://t09468ns.wordpress.com/2013/04/16/%E3%82%BD%E3%83%95%E3%83%88%E3%82%A6%E3%82%A7%E3%82%A2%E5%B7%A5%E5%AD%A6-%E7%AC%AC%EF%BC%92%E8%AC%9B/

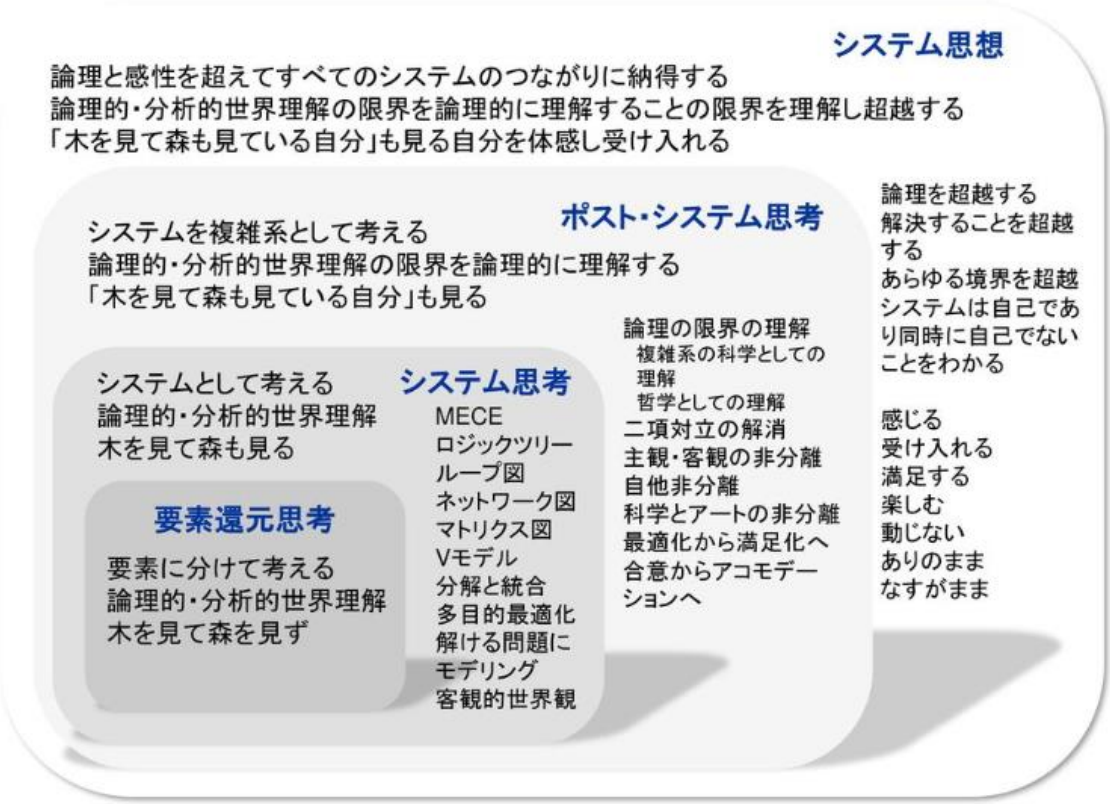


参照：

https://t09468ns.wordpress.com/2013/04/16/%E3%83%A2%E3%83%87%E3%83%AB%E3%81%AB%E5%9F%BA%E3%81%A5%E3%81%8F%E3%82%B7%E3%82%B9%E3%83%86%E3%83%A0%E3%81%AE%E4%BA%88%E6%B8%AC%E3%81%A8%E5%88%B6%E5%BE%A1%E3%80%80%E7%AC%AC%EF%BC%92%E8%AC%9B/

⑨ エンジニアリング分野を含む、システムの科学と哲学の体系がある。下図は、いわゆる木（要素）から森（システム）、複雑系へと広がり、思想にまで展開される構図を示している。ただし、工学の領域は、システム思考までを考えればまずは十分と思われる。

システム関連の用語として、システミックとシステマティックがある。システミックとは「システム」としてという意味で、ホリスティック（全体として）、俯瞰的に、体系的に、という言葉に近い。下図では、ポスト・システム思考に近い。システマティックとは、システムを要素還元論的に分解して理解するとともに、分解された要素を再構成してシステム全体も理解するというやり方であり、システムズ・エンジニアリングの基本はシステマティックであるということである。下図では，システム思考に相当する。



参照：http://slidesplayer.net/slide/11341170/

編集後記

今回の委員会の参加前には SoS やシステムズ・エンジニアリングという言葉や概念を理解しておらず、システムズ・エンジニアリングは航空宇宙など大規模システムの設計などに使われるもので、一般産業界にはあまり浸透していないのではないか、またこれから IoT などでシステム志向が高まる中で、SysML は主にソフト系で展開されているが、機械・電気・ソフト系の共通言語として今後有効に活用できるのではないか、と漠然と考えていた。そしてメカトロニクス系のメーカーでシステムズ・エンジニアリングを浸透させるにはどうすればよいかを考える機会になると思い、委員会に参加した。

ところが、システムズ・エンジニアリングは SysML などのツールを使って学ぶことから始めるではなく、もっと本質的な概念を学ぶことが重要であることを初回の委員会で多くの方から指摘を受けた。そして各委員から成功や失敗の取り組み事例が紹介され、既に各社でシステムズ・エンジニアリングが展開されていることを知った。さらに展開する際には大学の先生方に指導頂くなど、外部の知恵を活用していることもわかった。委員会の期間中には MSTC 様主催の慶應大学 SDM 研究科・白坂教授の講演会が開催され、システムズ・エンジニアリングの考え方を学ぶことができたことも大きな収穫であった。システムズ・エンジニアリングという言葉ではないが、一つの実用的な解がマツダ社の機能エンジニアリングであり、スカイアクティブエンジンの成功に結びついたことも知ることができた。

システムは複数のサブシステムから形成されるため、システムがつながっても基本的にはシステムになるが、IoT 時代には独立かつ個別管理されているシステム同士がつながるため、SoS という概念が非常に重要になってくる。今回の委員会では時間的な制約を受けてガイドライン策定までとなったが、次はソフト系から発展させてメカトロニクス系でシステムズ・エンジニアリングを展開するにはどうすればよいかの実践的な手引きがあれば、もっと考え方が広まるのではないかと考えている。

メンバー、委員会開催記録

【メンバー】

主査

首藤　俊夫	㈱三菱総合研究所

委員

今成　宏幸	東芝三菱電機産業システム㈱
益　啓純	㈱ジェイテクト
後藤　宏二	三菱電機㈱
小林　泰山	富士通㈱
畑　義裕	村田機械㈱

オブザーバー

島田　昌典	㈱三菱総合研究所
水上　潔	ロボット革命イニシアティブ協議会
北村　篤史	ロボット革命イニシアティブ協議会
益子　龍太郎	ロボット革命イニシアティブ協議会

【委員会開催内容】

第 1 回 10/11　メンバー紹介、問題意識の共有
第 2 回 12/12　システム化の課題事例の紹介（１）
第 3 回 1/18　システム化の課題事例の紹介（２）
第 4 回 2/08　ガイドブックの考え方
第 5 回 2/26　ガイドブックの構成内容（１）
第 6 回 3/15　ガイドブックの構成内容（２）
第 7 回 3/30　ガイドブックの記載内容