

[デジタルアーキテクチャデザイン]

② 社会・産業アーキテクチャのデザイン



白坂成功

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

アーキテクチャとは

「アーキテクチャ」という言葉を聞くと、多くの人はIT アーキテクチャやソフトウェアシステムのアーキテクチャを思い浮かべる人が多いと思う。もちろん、元々の意味である建築物としてのアーキテクチャを考える人もいるだろう。あるいは近年は、ビジネスアーキテクチャという言葉も使われているため、聞いた方もいると思う。このようにアーキテクチャという言葉だけでは、何を意味するのかが明確でない。元々システムアーキテクチャという言葉は、対象とするシステムがあり、そのアーキテクチャを指すものである。ここでは、その定義を明確にすることから始める。

システムとシステムアーキテクチャ

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 (element) を組み合わせたものである。これにはハードウェア、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」¹⁾とされている。つまり、システムの対象は、ハードウェアやソフトウェアのみではなく、人が相互に作用する組織やコミュニティ、組織や人が相互に作用するビジネス、そしてそれらすべてが相互に作用する社会なども含まれる。つまり、システムアーキテクチャとは、システムとして捉える対象とする範囲があり、その対象範囲のアーキテクチャをシステムアーキテクチャと呼ぶ。

また、アーキテクチャとは「システムが存在する環境の中での、システムの基本的な概念または性質であって、その構成要素、相互関係、ならびに設計および発展を導く原則として具体化したもの」^{☆1}とされている。つまり、システムの持つ定義された目的を達成するための基本的構造であり、その具体的な構成要素と構成要素間の相互関係やその原則のことである。

社会・産業アーキテクチャ

社会・産業アーキテクチャとは、社会や産業をシステムとして捉える対象として、そのアーキテクチャを社会・産業アーキテクチャと呼ぶ。社会や産業には、ハードウェアやソフトウェアのようなものだけでなく、サービスやビジネス、人々や企業、あるいはルールなど、多様な範囲が含まれる。つまり、対象とする社会や産業があれば、その社会や産業が持つ目的を実現するための基本的構造であり、どのような要素から構成されているのか、また、それらはどのような相互関係があるのか、そしてそれらの原則はどんなものかを示したものである。

なぜアーキテクチャなのか

これまでも社会・産業は存在していた。またさまざまな取り組みが社会・産業に関して行われてきた。なぜ今、アーキテクチャに着目することが重要な

☆1 日本規格協会, JIS X 0170:2020 システムライフサイクルプロセス (2020).

か。ここではその重要性の原因として考慮すべき3つの点を説明する。

新しいタイプのシステム

これまでのシステムは、ITシステムのようなソフトウェアが中心となっているものや、飛行機やロケットなどのようにハードウェアが中心となっているものが多かった。しかし、現在のシステムでは、もっと幅広く考える必要がある。日本政府が推進する Society 5.0 は、つながる社会を実現するものである。Society 5.0 では、これまでつながっていなかった分野同士がつながり、サイバーとフィジカルを融合することで、人々にとっての新価値を創造することを目指すものである。それぞれ独立して調達・運用・管理されるシステムが繋がって新たな価値をうむものを System of Systems (以下、SoS) と呼ぶ。たとえば、スマートグリッドやインダストリー 4.0、スマートシティ、IoT システムなどは SoS の代表例である。近年 SoS がますます社会で重要な役割を担うようになってきた。たとえば、スマートシティは、自治体単位で独立して構築・運用・管理されるものであるが、利用者は、それらをまたがって利用する。つまり、東京から神奈川に移動したときに、スマートシティの仕組みに連続性がないと利用者からすると利用がしづらくなってくる。このように、SoS では、システム間の相互運用性 (Interoperability) が重要となってくる。単にシステムとして成立するだけでは不十分で、ほかのシステムの相互運用性を考慮する必要がある。これも Society 5.0 のようにつながる社会を目指す場合には重要なポイントである。

欧州標準化委員会 (CEN)、欧州電気標準化委員会 (CENELEC) および欧州電気通信標準化機構 (ETSI) は、スマートグリッドにおいて相互運用性をマネジメントするために、スマートグリッドリファレンスアーキテクチャとして Smart Grid Architecture Model (SGAM) を作成した。SGAM

は、ドメイン軸、ゾーン軸および相互運用レイヤから構成される 3 次元構造となっている。特に、相互運用レイヤはビジネスレイヤ、機能レイヤ、情報レイヤ、通信レイヤ、および機器レイヤから構成される。この SGAM を利用することで、スマートグリッドを構成する機器間の相互運用性を実現する²⁾。この SGAM を元にインダストリー 4.0 にあわせて作られたものが、インダストリー 4.0 リファレンスアーキテクチャモデル (RAMI4.0) である³⁾。RAMI4.0 も SGAM と同様に 3 次元構造となっている。これら 2 つの例にある通り、SoS の相互運用性を確保するために 3 次元構造のリファレンスアーキテクチャをつくるというものが生まれてきた。ただし、リファレンスアーキテクチャ自体は、色々な目的で利用されるため、必ずしも相互運用性のために使われるものではないことを注意したい。これらの知見も踏まえ、日本の Society 5.0 でも図-1 に示される Society 5.0 リファレンスアーキテクチャが作られた。

上記 2 つのモデルと同様に相互運用性を目指すためのものであるが、その範囲が大きく異なっている。図-1 から分かる通り相互運用レイヤに「戦略・政策」、「ルール」、「組織」など、これまでの相互運用性ではあまり扱ってこなかった項目が含まれている。

具体的にスマートシティを例に考えてみる。たと

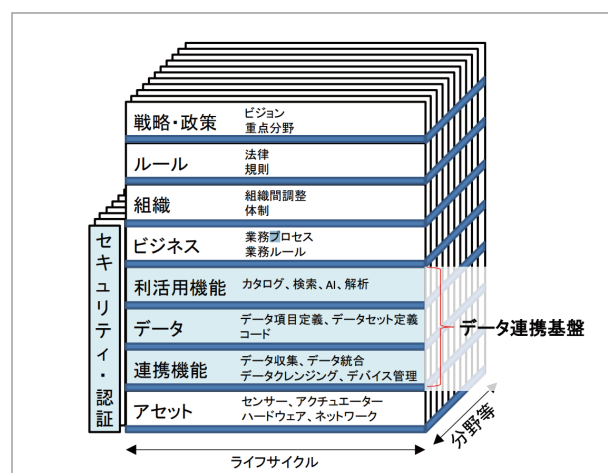


図-1 Society 5.0 全体アーキテクチャ⁴⁾

えば、通常、システムのインターオペラビリティというと、データフォーマットを合わせることや、通信プロトコルを合わせるなどがすぐに行われる。もちろんこれらの活動は重要である。しかし、それ以外の項目でも相互運用性が必要となる。たとえば、加古川市は、スマートシティの事例としてよく取り上げられる。加古川市では、多くの見守りカメラが設置されている。このカメラを利用して、防犯だけでなく、子供や認知症患者の見守りも検討されている。もし仮にこれらの目的のために隣の自治体が連携しようとカメラを設置し、フォーマットやプロトコルを合わせてもそれだけで相互運用可能になるわけではない。なぜなら、加古川市は、カメラを設置・利用するために市民対話を繰り返し、条例をつくり、パブリックコメントを受けることでこのカメラを使えるようにしている。同様のルールを持たない自治体と相互運用をするのは簡単ではないと考えられる。つまり、隣の自治体と条例の相互運用性をデザインする必要がある。では、隣の自治体と調整すればよいのであるが、そのためには都市マネジメントの主体が自治体である必要がある。実際には、スマートシティ推進組織には3つのパターンがある。「自治体主導モデル」、「民間主導モデル」および「地域協議会主導モデル」である⁵⁾。この推進体制が異なると、調整機能を働かせるために一工夫が必要となる。つまり、推進組織の相互運用性もデザインする必要があるということになる。

環境の急激な変化

次に、環境の急激な変化への対応の必要性があげられる。VUCAという言葉をご存知だろうか？ VUCAとは、Volatility（変動）、Uncertainty（不確実）、Complexity（複雑）、Ambiguity（曖昧）から構成された言葉である。現在の世の中はVUCAの時代と呼ばれ、先の予測ができない、計画通りにならない時代であると言われてきた。また世界不確実性指数も増加の一途を辿っている。つまり、将来

が不確実な時代になってきたということが共通認識になっている。COVID-19はまさにそれを決定づけた一例であると言える。

しかし、単純に世の中の変化が激しくなっただけではなく、我々自身が世の中の変化を受けやすい社会をつくっている。上述の通り、日本政府が推進するSociety 5.0は、つながる社会を実現するものである。Society 5.0では、これまでつながっていなかった分野同士がつながり、サイバーとフィジカルを融合することで、人々にとっての新価値を創造することを目指すものである。つながる社会は変化の影響範囲を広げる物である。つながっていない2つのシステムで構成される場合、どちらか一方のシステムに影響を与える環境変化が起きても、影響を受けるのはそのシステムに限られる。しかし、これら2つのシステムがつながっていると、どちらか一方だけに影響を与える環境変化であっても、他方のシステムにその影響が伝播する可能性が増加する。つまり、つながる社会は、世の中の変化の影響を受ける可能性が増加することになる。

環境が急激な変化をする場合、次のようなことが発生する。まず開発期間中に環境が変化すると、開発の想定が変わってしまう。また、運用中に環境が変化すると、これまで正常に動作していたシステムが、そのシステム自体には問題がないにもかかわらず、正常に目的をはたせなくなってしまう。

通常のシステムは、外部との関係が変化しないと仮定したクローズシステムとして扱う。しかし、このように外部との関係が変化すると仮定したものをオープンシステムと呼ぶ。

オープンシステムのデザインにおいては、2つのことを考慮しなければならない。1つは、環境の変化がシステムに影響を与えることに気がつく必要がある。もう1つは、変化に対応するために、変化に対応しやすいシステムデザインをすることである。

たとえば、前者の1つのアプローチとして、「一般社団法人ディペンダビリティ技術推進協議会（略

称 DEOS 協会)」では、設計時に設計者がどのような外部環境変化が設計に影響を与えるかを明示化するための手法として D-Case を提案している。D-Case を活用することで、ステークホルダ間で要求の変化に対する合意議論を十分に行うことができ、合意結果／結論に至った理由／議論の経緯を記録することができる。

後者のアプローチとして、変化を考慮したアーキテクチャ設計が挙げられる。たとえば、自動運転車を考えてみる。今は、“カメラで信号機を読み取る”ことによって、信号を判断する必要がある。しかし、将来的には、インフラが整うと、車—インフラ間通信で信号情報をインフラから受け取ることが可能となるかもしれない。このとき、どちらも“信号の状態を知る”という機能では同じである。このとき、手段が“カメラ”から“通信”に変わる可能性があることを考慮して、“信号の状態を知る”という機能は変わらないという前提でアーキテクチャを設計しておけば、すべてをやり直す必要はなく、設計を変更可能となる。

これらのように環境が変化するというリスクについては、環境の変化がシステムに影響を与えることに気がつく仕組みを導入することと、変化に対応するために、変化に対応しやすいシステムデザインをすることで対応が可能となる。

2020 年のものづくり白書では、このように変化に対応可能な組織の能力を持つことこそがダイナミックケーパビリティの本質であり、DX はその能力を高めることに貢献できることであると述べている。

重要なシステム特性と説明責任の増加

3 つ目として、重要なシステム特性と説明責任の増加について説明する。マサチューセッツ工科大学がシステム特性に関して行った調査によると、近年新たなシステム特性に関する論文が出て、さらにそれらの本数が急激に増加していることを示してい

る⁶⁾。システム特性とは、システムとして捉えないと評価ができない特性のことであり、古くは品質 (Quality) や安全性 (Safety) などがこれにあたる。近年では、前述した相互運用性 (Interoperability) や総合信頼性 (Dependability) などその種類は増加しつづけている。つまり、環境の変化が激しく、システムの範囲が拡大しているにもかかわらず、システムとしてしか捉えられない特性が増えているということになる。

さらにこれらに対する説明責任が増加してきている。たとえば、IEC 61508 や ISO 26262 のような機能安全規格は、安全性の説明責任を要求している。鉄道の機能安全規格である IEC 62278 は、RAMS という名前のとおり Reliability (信頼性)、Availability (可用性)、Maintainability (保守性)、Safety (安全性) の 4 つのシステム特性の説明責任が求められている。現在、色々な対象において、機能安全規格という名前のもと、安全性およびその他のシステム特性の立証を求める規格がつけられている。自動車の ISO 26262 などのものであるが、これまで求められていなかったものが求められるようになる、規格へ対応しなければならなくなる。しかしながら、この機能安全規格をはじめとしたシステム規格は、開発当初からトップダウンで考慮しなければならないため、これまでの製品が、製品としては問題がなくても、立証ができないということが起きてしまう。

今後のシステム規格への対応を考えると、あらゆるシステムの開発において、開発当初から、トップダウンで重要なシステム特性を設計し、その立証のためのエビデンスを残すというプロセスの規定と、そのための整備を行っておくことが必要である。日本人はトップダウンで行うリスクベース／ゴールベース開発が苦手であるということがよく言われるが、筆者は決してそうは感じていない。確かに、これまで必要がなかったため、そのような開発が行われてこなかった。このため、急に対応が難しいとこ

ろはある。しかし一方で、必要性が生まれて、きちんと考え方を学び、それを実践することをはじめている会社では、徐々にではあるが、そういった開発を行っている。つまり、きちんと学び、実践をしていけば、そういった開発アプローチができる人材は日本でも育成が可能であるということである。

アーキテクチャデザイン

システムのアーキテクチャをデザインするアプローチは標準としても存在する。また、特定の対象範囲をシステムとして捉えたときのアーキテクチャデザインを容易にするためのアーキテクチャフレームワークが存在する。しかしながら、社会・産業をシステムとして捉えたときのアーキテクチャをデザインするための具体的なアプローチはまだ存在していない。ここでは、システムアーキテクチャのデザインについて述べたのち、今後、社会・産業アーキテクチャをデザインするために重要な3つの点を述べる。

システムアーキテクチャデザイン

システムを構築するための方法論として、システムズエンジニアリングというものがある。システムズエンジニアリングとは、「システムの実現を成功させることができる複数の専門分野を束ねるアプローチおよび手段」¹⁾と定義される。つまり、システムズエンジニアリングとは、複数の専門分野(たとえば、電気工学、機械工学、ソフトウェア工学など)を統合し、束ねるためのアプローチである。なんらかの課題を解決したり、価値を創造することは、単一の専門分野だけで実現することは難しく、複数の専門分野の統合が必要となることを考えると、課題解決や価値創造のために必要は基本的な考え方であるといえる。ISO 15288では、テクニカルプロセスの中に、アーキテクチャ定義プロセスとしてアーキテクチャデザインが規定されている。アーキテ

クチャデザインプロセスの詳細は、ISO 42020として定義されており、その表記法はISO 42010、その評価法はISO 42030として規定されている。つまり、システムアーキテクチャのデザインについては一般的なレベルとしては明確となっている。また、システムの範囲を限定すると、その範囲の特徴にあわせたアーキテクチャデザインを支援するためのフレームワークが存在する。たとえば、エンタープライズシステムについては、エンタープライズアーキテクチャのフレームワークとして、米国のFederal Enterprise Architecture Frameworkをはじめ、The Open GroupのThe Open Group Architecture Framework (TOGAF)など多くが存在する。また、米国国防総省は、武器システムの調達のためにDoD Architecture Framework (DODAF)を作成し、それへの準拠を求めている。それ以外にも鉄道システムのためのThe Rail Architecture Framework (TRAK)や、人工衛星のためのThe European Space Agency Architecture Framework (ESA-AF)など多様なものが存在する。このようにいわゆるハードウェアやソフトウェアのシステムのアーキテクチャについては、そのデザインのアプローチについてそれなりに確立していると言える。

社会・産業アーキテクチャデザイン

社会・産業アーキテクチャデザインについては必ずしも確立しているとはいえない。それは、図-1のSociety 5.0の例で示した通り、その範囲が広く、これまでのやり方やアーキテクチャフレームワークでは明らかにカバーされていない範囲が存在するからである。そこで、対象をシステムとして捉えてデザインするという根本的なところに立ち戻って考える必要がある。ここでは重要な3点を説明する。

①多視点から見る

対象をシステムとして捉えるためには、対象を多視点から見て考えることが重要である。対象によって最適な視点は異なるが、一般的に役立つ視点があ

る。それは、時間の視点、空間の視点、意味の視点（機能の視点、物理の視点）である。時間の視点は、対象となるシステムの観点ではライフサイクルと呼ばれ、ユーザの観点ではカスタマージャーニーなどと呼ばれる。機能と物理は別の視点であることを認識して、分離して考えるだけで有益である。また、多様な分野をデザインするためには多様な視点から捉えることが必要となる。

②俯瞰的に捉え、体系的に考える

多視点で見たものを、それぞれの視点ごとに俯瞰的に捉え、それらの要素と要素間の関係性を体系的に考えることが必要である。たとえば、時間の視点で見て、俯瞰的に捉えることでライフサイクル全体を視野にいれ、体系的に考えることで、ライフサイクルをもれなく分割して系統的に考えることがこれにあたる。

②抽象度をコントロールする

ある視点から見て、俯瞰的に捉え、体系的に考えるときには、抽象度をコントロールすることが重要である。人は、あまり多くの数を一度に把握することが難しい。このため、いきなり全体を 20 や 30 に分割するのではなく、まずは 5 つに分割し、それぞれをさらに 5 つに分割する。こうすれば、最終的には全体を 25 に分割することになるが、全体感を失

わないように徐々に細部を考えていくことが可能となる。

今後は、社会・産業アーキテクチャデザインの実践を積み重ね、具体的な方法論を構築していくことが重要である。情報処理振興機構に設置されたデジタルアーキテクチャ・デザインセンターにおいてこれらの知見がたまとともに、方法論が構築されることが期待される。

参考文献

- 1) INCOSE Systems Engineering Handbook : A Guide for System Life Cycle Processes and Activities Ver. 4.0 (2015).
- 2) 齊藤 健, 正畑康郎, 大場義洋: スマートグリッドの最新の標準化動向と東芝の取組み, 東芝レビュー, Vol.68 No.8 (2013).
- 3) 日本貿易振興機構: インダストリー 4.0 実現戦略調査報告書 (翻訳版).
- 4) 内閣府: 分野間データ連携基盤の整備に向けた方針案 (2018).
- 5) 内閣府: スマートシティリファレンスアーキテクチャホワイトペーパー (2020.3).
- 6) Olivier L. de Weck : Engineering Systems : Meeting Human Needs in a Complex Technological World, The MIT Press (2011).

(2021 年 2 月 1 日受付)

■ 白坂成功 (正会員) shirasaka@sdm.keio.ac.jp

1994 年東京大学大学院工学系研究科修士課程修了。2011 年慶應義塾大学大学院 SDM 研究科修了。博士 (システムエンジニアリング学)。2010 年より慶應義塾大学大学院 SDM 研究科 准教授。2017 年同教授。大規模システム等の開発方法論の研究に従事。

