

DX 実践手引書

IT システム構築編 完成 第 1.0 版



IPA

独立行政法人 情報処理推進機構
社会基盤センター

このドキュメントは下記の URL からダウンロード可能です。

https://www.ipa.go.jp/ikc/our_activities/dx.html#section7

目 次

はじめに.....	6
第1章 DXを実現するための考え方.....	9
1.1. DXの起点は「目指すべきビジョン」の共有.....	9
1.2. 挑戦しやすい組織環境.....	12
1.3. 企業経営の中核課題となる内製開発力の強化.....	15
1.4. DXの実践を支える人材.....	17
1.5. アジャイルマインド.....	19
第2章 DXを継続的に進めるための考え方.....	22
2.1. 変革規模と組織成熟度の定義.....	23
2.1.1. 変革規模の考え方.....	23
2.1.2. DXの実現に必要な組織成熟度.....	24
2.2. DXを継続的に進めるための有効な施策.....	26
2.2.1. トップダウンとボトムアップの両面から火をつける.....	26
2.2.2. 活動を全社に波及させるための身体作りを行う.....	30
2.2.3. デジタルカンパニーの流儀を体得する.....	35
2.3. DXを継続的に進めるための考え方のまとめ.....	40
2.4. DXの実践における課題の対応事例.....	41
第3章 DXを実現するためのITシステムのあるべき姿.....	56
3.1. 「社会最適」を実現するためのITシステム要件.....	58
3.2. 「データ活用」を実現するためのITシステム要件.....	61
3.3. 「スピード・アジリティ」を実現するためのITシステム要件.....	69
3.4. DXを実現するITシステムのあるべき姿の全体像.....	74
第4章 あるべきITシステムとそれを実現する技術要素.....	76
4.1. DXを実現するための技術要素群の全体像.....	76
4.1.1. あるべきITシステムを実現する技術要素群の全体構成.....	76
4.1.2. あるべきITシステムを実現する技術要素群内の「組織内サービス」の各要素.....	77

4.1.3. あるべき IT システムを実現する技術要素群内の「外部サービス」の各要素.....	79
4.1.4. 「組織内サービス」の各要素におけるセキュリティの考え方.....	80
4.1.5. あるべき IT システムとクラウドの関連.....	82
4.1.6. あるべき IT システムと IoT の関連.....	84
4.2. アジャイル開発.....	88
4.2.1. アジャイル開発が必要とされる背景.....	88
4.2.2. アジャイル開発とは何か.....	89
4.2.3. アジャイル開発により得られる効果.....	96
4.2.4. アジャイル開発の考慮点.....	97
4.2.5. アジャイル開発の先進取り組み事例.....	98
4.3. スピード・アジリティを支えるマイクロサービスアーキテクチャの活用.....	103
4.3.1. マイクロサービスアーキテクチャを採り上げる背景.....	103
4.3.2. マイクロサービスアーキテクチャの活用により実現できる内容.....	106
4.3.3. マイクロサービスアーキテクチャを活用する上での考慮点.....	106
4.3.4. マイクロサービスアーキテクチャの活用事例.....	111
4.4. データ活用.....	116
4.4.1. データ活用のための環境整備のプロセス.....	117
4.4.2. データ活用を進めるために必要となる要素.....	119
4.4.3. 要素ごとの考慮点.....	125
4.4.4. データ活用の事例.....	136
4.5. API.....	139
4.5.1. API の概要.....	139
4.5.2. API 技術利用の背景.....	139
4.5.3. API の活用により実現できること.....	140
4.5.4. API の活用時の考慮点.....	141
4.5.5. スサノオ・フレームワークとの関連.....	144
4.5.6. API 全体管理とスサノオ・フレームワーク領域別の特徴.....	146
4.5.7. API 活用の事例.....	154
4.6. IoT.....	165
4.6.1. IoT の概要.....	165
4.6.2. IoT 活用の背景.....	165
4.6.3. IoT の活用により実現できること.....	167
4.6.4. デジタルツイン.....	168
4.6.5. 製造業での IoT の活用事例.....	170
4.7. 社会最適を実現するための外部サービスの活用.....	173
4.7.1. 社会最適とは.....	173
4.7.2. 割り勘効果.....	173
4.7.3. 割り勘効果が期待できる外部サービスの種類と特徴.....	174
4.7.4. 外部サービス利用のメリット・デメリット.....	175

4.7.5. 競争領域、非競争領域と外部サービス	176
4.7.6. デジタル産業の企業4 類型との関係.....	178
4.7.7. 競争性の変化	180
4.7.8. 外部サービス活用方法のまとめ	182
4.8. 現行システムからあるべき姿への移行.....	183
4.8.1. 移行の考え方	183
4.8.2. 段階的移行の方法.....	184
4.8.3. 段階的移行により実現できる内容	185
4.8.4. 段階的移行に関する考慮点	186
4.8.5. 段階的移行の事例	187

はじめに

あらゆる産業において、新たなデジタル技術を使ってこれまでにないビジネスモデルを展開する新規参入者が登場し、実際にゲームチェンジが起こっている。こうした中で、各企業は、競争力維持・強化のために、デジタルトランスフォーメーション（DX：Digital Transformation）をスピーディに進めていくことが求められている。日本における DX は、2018 年に経済産業省が「デジタルトランスフォーメーション（DX）を推進するためのガイドライン¹」を発表したことを契機に広がりを見せた。そこでの DX の定義は「企業がビジネス環境の激しい変化に対応し、データとデジタル技術を活用して、顧客や社会のニーズを基に、製品やサービス、ビジネスモデルを変革するとともに、業務そのものや、組織、プロセス、企業文化・風土を変革し、競争上の優位性を確立すること」であり、企業がグローバル競争で今後勝ち残ってゆくために必須の取り組みであるとしている。

この背景には、センサー技術や IoT の発達、データ通信容量の増大などにより、現実世界で発生する事象がデータ化され、さらに大量のデータの処理を可能にする計算機性能の向上や大量のデータから法則性を発見するソフトウェアが登場したことがある。これらのデータを取得し、活用することで、新たなビジネスモデルや価値を生み出す企業が 2000 年代から次々と生まれた。代表的な例として「GAFA²」や「BATH³」といった括りで表現されるような企業群によって、これまでに存在しなかった新たな製品やサービス、ビジネスモデルが出現するようになった⁴。このような企業群のように新たに市場を創出する企業、創出した市場に参入するプレイヤー、市場そのものを変化させる「デジタルディスラプター（先進的なデジタル活用による既存の秩序やビジネスモデルを破壊する企業）」によって、従来のビジネスモデルが破壊される事例も相次いだ。さらに重要なのは機能分解された複数の業種・業態のビジネスをリバンドリングすることによって、今まででなかった市場を創造する動きが起きていることである。このリバンドリングによる市場創造の動きの中で主導権を握っていくためには、あらゆる産業において、企業はデジタル技術を用いて業務変革やビジネス変革するだけでなく産業領域間をデジタル技術の活用によりうまくつなげていくことを意識せざるを得ない状況にある。

このようにビジネスの中心や利益の源泉がデジタルの世界に急速にシフトする激しい変化の中で企業が競争力を保ち、生き残るには、デジタル活用を通じて社会や顧客のニーズに応え、サービスを提供する企業（デジタルエンタープライズ）になることが求められている。このための

¹ 経済産業省 デジタルトランスフォーメーションに向けた研究会,「DX レポート～IT システム「2025 年の崖」の克服と DX の本格的な展開～」, 2018 年 9 月, https://www.meti.go.jp/shingikai/mono_info_service/digital_transformation/pdf/20180907_03.pdf

² Google、Amazon、Facebook、Apple の米国企業 4 社の総称として知られる。

³ Baidu、Alibaba、Tencent、Huawei の中国企業 4 社の総称として知られる。

⁴ JETRO/IPA New York,「アメリカにおけるデジタルトランスフォーメーション(DX)の現状」, 2020 年 9 月, https://www.jetro.go.jp/ext_images/_Reports/02/1fb13cf2232a86ac/202009.pdf

一連の取り組みが DX を推進する原動力となっている⁵。

ただし、これからのデジタルエンタープライズに求められることは、先鋭的なディスラプターになることを目指すことに限らない。デジタル技術の活用を当たり前のものとして、自社の強みを生かしながら、異なる部門・企業・業界と連携する形であっても、スピーディに新しい価値を生み出すような変革を推進、継続することで成長の途は見出せる。そのためには、企業には、目まぐるしく変わりゆく事業環境に対して、俊敏に適応し、ビジネスプロセスのサイクルを高速化させるような仕組みづくりのみならず、技術やサービスを組み合わせた顧客体験価値の向上、または自らの強みを生かした社会課題の解決や社会変革への挑戦が求められているというべきである。

しかし現実には、我が国産業界における DX の推進は「DX の取り組みを始めている企業」と「まだ何も取り組めていない企業」に二極化しつつある。経済産業省は、企業の自己評価のための DX 推進指標を発表し、活用を呼びかけている。この取り組みにより提出されたデータを IPA が集計した結果⁶が公開されている

これによると、DX 推進の先行企業（レベル 3 以上）の割合は 2020 年度の調査で全体の 8.5% と、前年度の 4.4% よりも増加している。しかし、裏を返せば、依然として 9 割以上の企業でレベル 3 未満となっており、DX にまったく取り組んでいないレベルにあるか、DX の散発的な実施に留まっているに過ぎない段階である傾向が明らかになっている。

本手引書は、そのような社会背景を踏まえ、これから DX の実践に向けて取り組み始める、もしくは取り組んでいる最中にある組織の DX 推進の担当者を主な読者と想定している。

なお、DX の実践にあたり経済産業省や IPA は様々な指標やツール、ドキュメントなどを提供している。たとえば「DX 推進指標」は、前述のようなまだ DX に取り組めていない企業が自社の現在地を把握するために活用ができる。また「プラットフォームデジタル化指標」は、自社の IT システムが DX の推進に適した状態かどうかを把握することができる。こうした指標を活用することで、自社の強み・弱みを把握することができる。是非、参考にいただきたい。

しかし、それでも実際 DX に取り組むにあたり、何から始めればいいのかわからないという DX 推進の担当者も多い。こうした指標などを用いて、自社の立ち位置を把握まで実施しても、その先 DX の実践、成功に向けてどのような取り組みを積み重ねていくべきなのか、そしてどのような技術を何のために活用していくべきのかまようこともあるだろう。本手引書は、DX の先進事例とされる企業の取り組みを調査し、そこから学び取れる教訓をまとめたものとして、そうした検討の中で DX 推進の参考となる情報を提供するものである。なお、この際の調査の概要については、IPA から報告書を公開している。⁷

⁵ 山本修一郎、「システム安全性向上一世界の最前線、9. デジタルトランスフォーメーションの課題」、月刊ビジネスコミュニケーション、2019 年 4 月号

⁶ 独立行政法人情報処理推進機構 「DX 推進指標 自己診断結果 分析レポート（2020 年版）」 2021 年 6 月
<https://www.ipa.go.jp/files/000091505.pdf>

⁷ 独立行政法人情報処理推進機構 「DX 先進企業へのヒアリング調査 概要報告書」 2021 年 10 月
<https://www.ipa.go.jp/files/000093364.pdf>

なお、本手引書は、DX について、これを支える技術的な側面からアプローチし、企業の DX 推進の担当者にとって必要になる考え方や IT システム構築における要件、技術要素を示すことを目的としている。

DX レポートでは、老朽化・複雑化・ブラックボックス化した既存システムが DX を本格的に推進する際の障壁となることに対して警鐘を鳴らすとともに、2025 年の完了を目指して計画的に DX を進めるよう促した。もとより、デジタルプラットフォームを整備することは、DX の一つのカギであること自体には変わりはない。しかし、先進事例の取り組みを調査すればするほど、DX の最終的なゴールはビジネスモデルや組織の変革であり、これを最終的なゴールとして継続的に変革し続けることであり、何かを「単にデジタル化すれば DX」、「技術的に刷新すれば DX」という誤解を避けたい、という考え方がいよいよ強くなった。

そのため、第 1 章では「DX を実現するための考え方」として、改めて DX とは何かという位置づけや技術を扱う以前に必要な DX の目的やその考え方を示すことから始めることにした。さらに、第 2 章では DX は一度実施すれば終わることではなく、継続して変革をし続けることが重要であることをふまえて、DX を継続的に進めるための考え方を示す。これらの考え方の中で、特に技術的な側面にフォーカスし、第 3 章、第 4 章を構成する。第 3 章「DX を実現するための IT システムのあるべき姿」では、DX を効果的に実現するための IT システムの目指すべき姿を示し、それらの特徴を解説する。第 4 章では、第 3 章で示された IT システムのあるべき姿に対して、技術要素ごとに具体的なアプローチと事例を紹介する。

DX レポートで扱った既存システムの刷新に関しては、本書の別冊として、『DX 実践手引書 IT システム構築編 レガシーシステム刷新ハンドブック』に現行 IT システムの全体把握や仕様復元の方法論を紹介しているので、合わせて参照いただきたい。

第1章 DX を実現するための考え方

DX を実現するうえで、デジタルプラットフォームとしての IT システムの更改や技術的ノウハウを取り入れることは重要である。しかし、IT システムやそれを支える技術そのものは、あくまでも市場環境で競争上の優位性を確立するための「手段」の一つであり、IT システムや流行りの技術的スキルを備えただけで完全に DX が実現されていると言うべきではない。DX の本来の目的は「競争上の優位性を確立する」ことである。そのために IT システムのみならず、組織ビジョンやビジネスモデル、組織風土を含め、組織のあらゆる面での変革を通じて、デジタルを前提とした付加価値の創出に取り組むことが重要である。

DX の取り組みでは IT システムや技術論のみに留まらず、デジタルの活用を前提とした、市場や社会、顧客、社員および業務プロセス等、新しい価値の創出に向けた継続的な活動が求められている。この点については、『デジタルトランスフォーメーションの加速に向けた研究会 WG1 全体報告書⁸』に記載されている課題を共有するものであるが、当該報告書では「DX の加速には経営層のマインドが重要であり、それらを社内外に発信・伝達するための対話が不可欠である（前述 WG1 全体報告書より引用）」という課題観点の位置づけから、主に経営層への提言をまとめた構成になっている。本手引書の想定とする主な読者は、これから DX の実践に向けて取り組み始める、もしくは取り組んでいる最中にある組織の担当者である。本章では、読者とする DX の担当者が IT システム構築以前に、DX に取り組む上で特に必要な要素を示している。その際に、どのような経営的認識や経営者視点の対応が必要かについては、WG1 の全体報告書を参照してほしい。

はじめに、IT システムそのものや技術論以前に、DX を推進するために組織内で共有すべき考え方は何なのか、先進企業の事例に共通して見られる事項をまとめる。

1.1. DX の起点は「目指すべきビジョン」の共有

全社的な取り組みとして DX を実践する第一歩は、まず社内の幹部を巻き込んで、組織の将来像・ビジョンを徹底的に議論し、デジタル活用を踏まえて、10 年、20 年先のビジョンを掲げて共有することである⁹。デジタルトランスフォーメーション (Digital Transformation) とは文字通り「デジタル変革」であり、組織全体としての変革に向けた力が必要になる。そうでなければ、多くの組織で DX を推進すべきであると頭ではわかっている、個々の行動に反映されず、他人事になってしまう。ビジョンの策定、共有がされていなければ、たとえそれぞれが良い取り組みを進めていても組織全体としての力を発揮できない。

⁸ 経済産業省 デジタルトランスフォーメーションに向けた研究会 WG1, 「デジタルトランスフォーメーションに向けた研究会 WG1 全体報告書」, 2020 年 12 月, <https://www.meti.go.jp/press/2020/12/20201228004/20201228004-4.pdf>

⁹ 事業や組織の変革を成功させる実践知を抽出し、パターン・ランゲージとして、24 のパターンで整理した「トランスフォーメーションに対応するためのパターン・ランゲージ」も、合わせて参照されたい。
<https://www.ipa.go.jp/files/000090310.pdf>

デジタル活用による革新や、その考え方を企業や組織の隅々まで浸透させ、ガバナンスをきかせていくことは容易ではない。各部門それぞれの個々の業務やサービスは、絶えず工夫を凝らしながら最適化されている。日々の現場の努力の積み重ねが外的な圧力により様変わりするとなれば、反発があるのは当然と言える。だからこそ、経営層を含めて、組織内外との対話を通じて、目指すべき大きなビジョンを軸として共有することが重要である。そしてそのビジョンを組織の構成員に浸透させるために、社内文化や人事制度などのマネジメント・ガバナンス、そして具体的な戦略に整合させてゆく。これはいずれの DX の先行事例にも共通して見られるプロセスである。

つまるところ、DX の実践の核とすべきは、「なぜデジタル化しなければならないのか」「デジタル化を進めて自分たちがどのようなになるのか」という未来に向けた問いに答えられる「ビジョン」である。なぜなら、このビジョンが中核にあることで、全ての取り組みや行動に共通する一貫した意図が関係者に伝わり、賛同・協力を得ることができるからである。ビジョンは組織の将来像や目指す姿、未踏の目標と言っても良い。それはおよそその経営方針や、いつまでにどれだけの利益を計上するといった中長期の計画ではない。例えば、「テクノロジーで何百万もの人々の生活を変える」(Apple 社)のような、社会全体の課題への貢献や、社会における企業の存在価値、未来の社会に与えるべき価値の提供内容を定義するものになっている。このビジョンを絵空事としないためにも、ビジョンの策定はできる限り自らの企業としての強みを土台にすることが望ましい。

次に、策定したビジョンを実現するためには、いつまでに何を達成し、今何をすべきなのかのロードマップを考えなくてはならない。このような、目標となる未来を起点に、現在を振り返る考え方を「バックキャスト」という。ビジョンを実現するための DX 実践へむけて、具体的な戦略や計画を立案し、いつまでに、どのような状態になっていきたいか、ロードマップ上の目標を考えるのである。これらの目標を一般的には「マイルストーン」と呼び、これは複数設定されることが望ましい。DX におけるロードマップやマイルストーンは一度決めたら普遍的で絶対に変えてはいけないというものではない。世の中の市場の変化や技術の変遷に合わせて、ビジネス上のニーズも時々刻々変化していくため、ロードマップやマイルストーンもそれに合わせて随時見直し、常に最適化・修正をしながらビジョンに向かっていくべきであろう(図 1.1.1)。

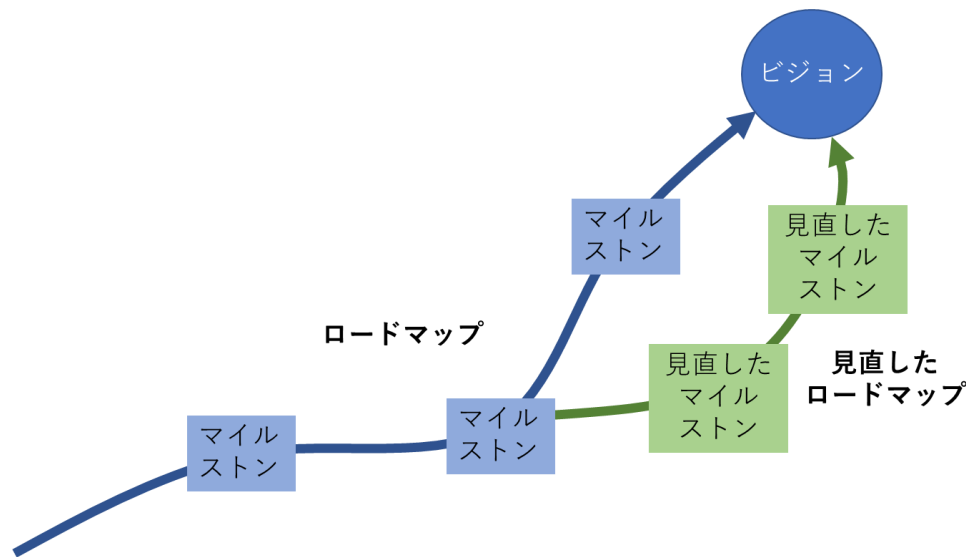


図 1.1.1 ビジョンとロードマップ

このようなビジョンやロードマップを経営層とも共有し、対話を繰り返す中で、その実現のためのコミットメントを経営層から引き出すことは、プロジェクトの担当者としては必ず経ておきたいプロセスである。人員・時間・予算などのコストがかかり、組織の未来を左右する取り組みとなる DX の実践において、経営陣のコミットメントが全く無いにもかかわらず変革に成功したというのは例がない。ビジョンやロードマップ、マイルストーンなどは組織全体で繰り返し共有されるべきものである。DX の推進自体は改革に熱意のある人材を集めた少数精鋭のチームで企画・計画し旗振りを行い制御する方がより効果的である場合が多い。そのような人材が意志を持って能力を最大限発揮させるためにも、経営者の強い意志とコミットメントを引き出すことが必要になる。この点は前述した『デジタルトランスフォーメーションの加速に向けた研究会 WG1 全体報告書』の中で経営層に向けた提言として、現場との対話の重要性が言及されている。

本手引書を作成するにあたり DX 先進事例の調査を行ったが、先進事例の中には、現場側から小さな成功体験、言い換えると、デジタル活用による業務の改善や顧客接点を増やすなどの社内の期待や信用を積み重ね、その現場側からの経営層への積極的な働きかけによって、経営層にデジタル活用の良さを実感してもらうような取り組みも見られた。この取り組みを重ねる中で、コロナ禍という危機的な状況に直面した際、急激なデジタル化への対応ができる組織の適応能力が予め備わっていたことにより、経営層が更なるデジタルの活用に注力するようになったという事例も見られた。

DX の実践において一定の成果をあげた先行事例企業においては、トップレベルでビジョンとそれに基づく戦略ロードマップを具体的に策定し、それを社内文化や人事制度等を通じて社員に浸透させる仕組みを作っている例が多い。策定されたビジョンが企業内に浸透して初めて事業がビジョンに向かって動き出すからであろう。例えばある会社では、経営層が合宿して作成したビジョンに基づいて、各事業責任者が自分の事業のロードマップ案を策定しこれを経営層と議論を繰り返すことで、ロードマップを確定する。ロードマップの各マイルストーン実現のためのプロジェクトには組織横断で手を上げることで参加できる仕組みを整え、手を上げることが昇

進等で評価される評価制度を整備していた。このように推進への寄与や積極的な行動なども人事制度と連動し評価に組み込むことは、個々の行動を促すためには最も有効な手段の一つとなりうる。大きな成果をあげている事例では、その浸透度が高く、組織内の幅広い人材の参加を促している。

また、業務プロセスにせよ、既存製品・サービスの変革にせよ、全く新しいビジネスモデルの創出にせよ、変革を実現するきっかけは、「アイデアの数とその品質・具現性次第である」という側面もある。そのため、ビジョンや目的、課題に取り組むべき理由を共有した組織の社員から、自発的にボトムアップな変革施策が自然発生してくるような状態が DX に取り組む組織の理想の状態といえる。その状態を実現するためにも、まずはビジョンを掲げ、共有・浸透することが DX の起点となる。

1.2. 挑戦しやすい組織環境

「DX の実践」と一口に言っても、その解釈は様々である。一般的に語られる DX の実践例には、「業務の効率化による生産性の向上」、「既存サービス・製品の高付加価値化」、「新規製品・サービスの創出」または「ビジネスモデルの抜本的な変革」「組織文化の変革」などゴールや方向性の異なる DX の取り組みが混在している¹⁰。また、DX レポート 2 では DX の取り組み領域を明らかにするために、各アクションの取り組み領域と DX の段階に分けて整理した上で、異なる DX フレームワークを紹介している¹¹。このように DX の実践には様々な形があり、どれも重要な取り組みであることは間違いないのだが、先行事例を見ると、目的や方向性が異なる取り組みはそれぞれチーム・体制を分けて取り組む傾向が強い。

先行事例の中には、CIO または CDXO が中心となって、ビジョンに基づいたビジネスモデルの抜本的な改革を旗印に、必要な「組織文化の変革」「業務効率化」「既存サービスの高付加価値化」を一貫して戦略的に執り行う事例も一部存在する。ただし、企業や組織の規模や構成、考え方にも左右されるため、これらはカリスマ性を持つリーダーがトップに立ち、全体構想から意思決定、実働までをトップダウンで進められた場合の特異な例ともいえる。

多くの事例では、CDXO などの責任者が社内の IT 投資・情報戦略の全体感を持って見られる体制を構築したうえで、「目的」に応じて順次グループを組成し、特定の分野における実質的な戦略的焦点を設定していた。このような場合においても、DX を進める責任者(CDXO)の掌握範囲は IT システムの範疇にとらわれるべきではなく、また DX 全般を掌握する権限を与えられていなければ、こうした組織の実現は難しいだろう。

¹⁰ 独立行政法人情報処理推進機構「デジタル・トランスフォーメーション推進人材の機能と役割のあり方に関する調査」, 2019 年 7 月, <https://www.ipa.go.jp/files/000073700.pdf>

¹¹ 経済産業省 デジタルトランスフォーメーションの加速に向けた研究会, 「DX レポート 2(中間とりまとめ)」, 2020 年 12 月, <https://www.meti.go.jp/press/2020/12/20201228004/20201228004-2.pdf>

	未着手	デジタイゼーション	デジタライゼーション	デジタルトランスフォーメーション
ビジネスモデルのデジタル化				ビジネスモデルのデジタル化
製品／サービスのデジタル化	非デジタル製品／サービス	デジタル製品	製品へのデジタルサービス付加	製品を基礎とするデジタルサービス デジタルサービス
業務のデジタル化	紙ベース・人手作業	業務／製造プロセスの電子化	業務／製造プロセスのデジタル化	顧客とのE2Eでのデジタル化
プラットフォームのデジタル化	システムなし	従来型ITプラットフォームの整備		デジタルプラットフォームの整備
DXを進める体制の整備	ジョブ型人事制度 リカレント教育	CIO/CDXOの強化 リモートワーク環境整備	内製化	

図 1.2.1 DX フレームワーク

さて、プロジェクトの実行可否の決定は、通常、CIO や CDXO などのトップレベルかそれに準ずる組織体制によって為されるが、実際にプロジェクトを進めるにあたっては、トップダウンのやり方だけではうまくいかない場合も散見される。これは、トップダウンで行うプロジェクトは、実行の判断に際して、想定しうる結果・利益に重きを置く傾向にあり、現場や市場のニーズが反映されにくいためである。また、異なるレイヤーが最終的な目的を共有せずにプロジェクトに関わり、場合によっては手段と目的を混同してしまうケースも多い。もしも、真に価値を生み出す変革を実現したいのであれば、プロジェクトに係るすべてのレイヤーが、IT 技術だけでなく自社ビジネスについて、それも現場のオペレーションや顧客体験のレベルで深く理解していることが必要である。現場の経験、課題から生まれた新たなアイデアを最大限活用するためには、プロジェクトの中で現場の意見が尊重される風土の醸成が不可欠である。

ここで重要なポイントが2つある。1つは、掲げられるビジョンにおいて、CEO や CIO、CDXO などの経営トップによる DX に対する強い情熱が示されていることである。ビジョンに情熱や危機感、強い想いが伴っていなければ、変革は生まれない。そして、このビジョンは全員が「腹落ち」している、つまり組織全体が納得して決めたものであることが重要である。経営は組織全体が決めたビジョンを繰り返し全体で確認し、問い直し、それを具現化するためアプローチするための具体施策に常に最適な投資判断をしなければならない。

もう1つは、「失敗を否定しない文化」を社内を作ることである。DX プロジェクトは先行事例が少なく、未知の部分が大きいため、その多くは失敗する可能性が高い。しかしながら、成功をつかむためには多くの失敗を乗り越えねばならないわけで、そのためにはアイデアの数が物を言う。社員が批判を恐れず、アイデアを考え、発信できるよう、日ごろからトライ＆エラーを繰

り返すことを経営者が推奨すべきである。

例えば、先行事例の多くの企業では、定期的に「アイデアソン」と呼ばれる、デジタル技術を活用した事業・サービス案や業務改善案に関する従業員による発表コンテストを開催するなど、ビジョンに沿った具体的なアイデアが現場の声としてボトムアップに経営に吸い上がる仕組みを有している。ここでは実現可否よりも、一見荒唐無稽に思われるアイデアをどうすれば実現できるか、顧客や社会にどのような新しい価値やインパクトを与えられるかという視点で考えることが重視される。変革を実現するには固定観念にとらわれず、これまでとは異なるルールや方法を発見することで、現在の延長線上にない場所にリーチすることを模索しなければならないからである。

たとえ実際に挑戦して失敗しても、それらを互いに否定し、立ち止まるのではなく、失敗を受け止め、そこから学ぶことで成功につなげていくことが重要な考え方である。そこでは、互いに「失敗を許容する」ことは大切な観点であるが、そこから何を学んで、次にどのように生かすかまでをどこまで突き詰めて考えられるかが、その後の変革への成功と失敗を分ける。そのためには、ビジョン実現のためのロードマップの上のマイルストーン毎に仮説を立て、その仮説が正しいか、間違っているのかをデータを集めて検証するというサイクルを実行し、ロードマップやマイルストーンを見直していくスタイルが重要である。そのためには、仮説を検証するために挑戦しその結果としての失敗を積み重ねるサイクルを短く回すことも重要である。このサイクルを回し続けるためには社内ルール、社内手続き、社内ツールの整備、そして、このような行動を促進する人事評価も整える必要がある。また、MVP キャンパスなどのフレームワークを活用し、各サイクルで何が実現できて、何が実現できなかったのかを明確にしておくことも有用である。

挑戦する際には、ソフトウェアを開発することも多いが、その際に重要なのは、次に生かすものは必ずしも作った成果物（システム）ではないことである。ソフトウェアを技術的負債化しないためにも、上記サイクルで得られた知見だけを活かして、作ったものを捨てる覚悟も時には必要である。

仮説と成功条件を明確にしたうえで、まずは挑戦すること、そのために挑戦できる組織環境が整備されていることが重要である。小規模でも複数の取組みを、できれば並行で進め、上手くない取組みがあっても、DX の活動自体を止めないようにすべきである。それらの並行する取組みは、一見全く関係のない取組みであっても、組織のビジョンを実現するためという共通した目的が根底でつながっていることが重要である。それによって、チーム・社員一人ひとりの繋がりを強化し、継続的に変革を起こすための挑戦ができる文化、そこから学び取り次の変革に生かせるようになる。だからこそ、ビジョンは組織・社員全体の共通の指針として、人の想いがのった同じ価値観が浸透していることが重要なのである。

1.3. 企業経営の中核課題となる内製開発力の強化

ユーザーや市場といった事業環境の変化の速度が急速に増している中、企業は、業務におけるデジタルサービスを、その品質を担保しながら、俊敏に対応させ続けなければ競争力を失いかねない。したがって、企業が DX を持続的に推進するためには、一定の内製開発力（社内開発力）を備えることが望ましい。

これまで多くの企業はデジタルサービスをもっぱら外部委託によって開発してきているが、外部委託開発はトライアルからのフィードバック、サービス実装者へのニーズの正確な伝達や設計実装などに時間的オーバーヘッドがかかりすぎるのが問題となる。特に、企業の競争優位性を確保すべき領域では、デジタルサービスの開発にもよい意味で市場の反応に基づいた試行錯誤も必要となり、一般的には、高いリリース頻度で俊敏に開発しながら、試行錯誤を繰り返していくことになる。従来型の外部委託によるシステム開発、運用のスタイルは、スピードの低下、費用の高止まり、取り組みの方向性の硬直化などの面で、プロセスに致命的な問題を生じさせやすい。

また、DX に取り組む上での課題を設定する際には、現場の人材の課題意識やニーズ、自社の強みを深堀することが重要であるが、それらの本質的な理解を外部の人材に求めることはそもそも難しいという点も留意すべきである。さらに、システムの開発を完全に外注頼りにしてしまうと、ロックインやブラックボックスの温床になり費用対効果を悪化させることも考慮すべきである。

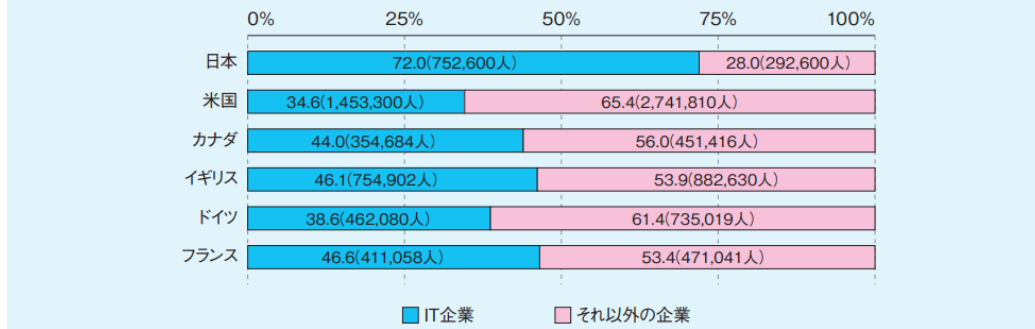
以上の理由により、内製開発力を強化することを、DX を持続的に推進するための礎として位置付けるものである。

しかし、内製開発力を強化するために必要な IT 人材の確保は現実問題として難しい。この理由の一つとして、そもそも日本の IT 技術者が少なく、かつその人材が IT 企業に集中し、さらに人材の流動性が低いことをあげてもよいだろう。

IPA 人材白書 2017 によると、日本以外の国は、IT 企業以外に所属する割合が 5 割を超えており、米国は IT 企業以外に所属する情報処理・通信に携わる人材の割合が 65.4%と最も高くなっている¹²のに対し、日本は IT 企業に所属する情報処理・通信に携わる人材の割合が 72%と突出して高くなっている。他の先進国と比較しても、日本の IT 技術者が IT 企業に偏って存在していることは明らかである。

¹² 独立行政法人情報処理推進機構, 「IT 人材白書 2017」, 2017 年 4 月, <https://www.ipa.go.jp/files/000059086.pdf>

図表2-2-3 IT企業とそれ以外の企業に所属する情報処理・通信に携わる人材の割合（日本、米国、イギリス、ドイツ、フランス：2015年、カナダ：2014年）³



1—情報処理・通信に携わる人材として扱った職種は、「システムコンサルタント・設計者」、「ソフトウェア作成者」、「その他の情報処理・通信技術者」
 2—情報処理・通信に携わる人材として扱った職種は、「11-3020 Computer and Information Systems Managers」、「15-1100 Computer Occupations」

3—以下を基にIPAが作成、カッコ内は情報処理・通信に携わる人材数

日本：2015年国勢調査結果（IT企業として扱った業種は、「ソフトウェア業」、「情報処理・提供サービス業」、「インターネット附属サービス業」。情報処理・通信に携わる人材として扱った職種は、「システムコンサルタント・設計者」、「ソフトウェア作成者」、「その他の情報処理・通信技術者」）
 米国：職業雇用統計（IT企業として扱った業種は、「511200 Software Publishers」、「518200 Data Processing, Hosting, and Related Services」、「541500 Computer Systems Design and Related Services」。情報処理・通信に携わる人材として扱った職種は、「11-3020 Computer and Information Systems Managers」、「15-1100 Computer Occupations」）
 カナダ：カナダ情報局（Statistics Canada <http://www.statcan.gc.ca>）のデータを基にICTCが作成（出典元「Digital Economy Annual Review 2015」）、情報処理・通信に携わる人材の職種、IT企業として扱った業種は米国の基準に準拠
 イギリス、ドイツ、フランス：EU統計局（Eurostat）が保有する労働力調査（EU LFS）の結果を調査会社であるempiricaが入手し分析したものを利用

図 1.3.1 IT企業とそれ以外の企業に所属する情報処理・通信に携わる人材の割合

本来このような状況下では、この差を是正するために必要なことはIT企業からの転職または派遣・出向などによる人材獲得を進めることが一つの方策となると考えられる。しかし、IT企業の給与水準は非IT企業のそれと比べて比較的高く、また、非IT企業にはそもそもどういった技術を持った人材にどのような役割を果たしてほしいかという職掌の定義をすることから困難を抱えているのが実情で、非IT企業にはIT企業からの人材獲得を進めることは簡単ではない。加えて、そもそも我が国のIT人材の総量の問題として、DXレポートでも2020年代後半にIT人材が43万人不足すると警鐘を鳴らしており、IT企業の側にしても非IT企業に転出させる人材的余裕はみられない（このことはIT企業の給与水準を押し上げる要因となり、非IT企業のIT人材獲得の障壁をさらに引き上げることになる）。

そこで、非IT企業自らがIT人材を育成する可能性が模索されるわけである。

とはいえ、十分な人的リソースや技術力の蓄積は、一朝一夕にして成るものではなく、これまで外部委託開発に依存してきたソフトウェア開発を非IT企業が自ら育成したIT人材ですべて内製化するという目標は、少なくとも短期的には、現実的ではないだろう。

そもそも、内製開発といってもいくつものレベルの業務に分解できる。すなわち、

- ① 自社で開発したものや、外部から導入したものなど、自社システムがどのように構成されているかの現状把握のレイヤー（CIO/CTOのロール）
- ② どのような事業目標をいかなるシステムの導入で、或いはシステムのどのような新規開発、機能拡充、変更で実現するか全体構想（グランドデザイン）、意思決定のレイヤー（PO（プロダクトオーナー）のロール）

- ③ 新規開発、機能拡充、変更に関する所要のシステム設計、コーディング（デバッグを含む）、リリース等の作業のレイヤー

であるが、このうち、③については、必要な人材量も多く、外部の IT 企業の力を借りることは合理的である。

但し、外部の IT 企業の力を借りるとはいっても、ここにも一定の戦略が必要ではないか。すなわち、企業の競争力の源泉となる価値を提供する領域やビジネス変化の速度が速い部分、さらにその中でも開発の方向性について不確実性が高い PoC（Proof of Concept）段階というべき部分について、あくまで外部の IT 企業に任せきりにせず、内製化された PO ロールの監督の下、社内人材を加えた混成チームとして運営する。信頼性・性能など品質を問われる部分は引き続き IT 企業の力を借りるとするのがまずは合理的な選択ではなかろうか。

技術者の育成には時間をかけた OJT を期待しなくてはならないが、このように IT 企業と共にシステム開発を進めていくことは、このプロセスを通して時間がかかる技術の学習、技術的目利き力の習得を実現し、長期的には広範囲な内製力の獲得にもつながると考える。

いずれにせよ、CIO/CTO ロール、PO ロールは自社内で確実に実施し、自社システムを自ら掌握することが必要である。

こう述べてくると、そこまでのことですら非 IT 企業には実現しがたいという声が聞こえてきそうである。しかしながら、これだけは飛び越えなければならない崖なのではないだろうか。

そもそも振り返れば、IT 人材の偏り・不足は、IT システムの「丸投げ」を続けてきた我が国産業界が自ら招いた構造的問題である。おそらく、そこにはここ 30 年の IT の発展に関する理解不足や、企業の業務プロセス改革に関する工夫不足が背景となり、IT システムに対して維持・管理をすれば十分であり、自社の強みを生かすようなプラスの投資対象であるという認識が乏しかったのではないかと考えられる。蓋し、もしそうでなければ、IT システムやデジタル技術が自社の事業のパフォーマンスに影響を及ぼすのならば、その投資は怠らなかったはずだからである。

しかし、DX に関する考察はこうした考え方が全く時代遅れのものになっていることを明らかにしている。したがって、もしも、この考えに賛同し、DX を推進していこうというのであれば、この長年の悪癖を改め、自らそのコストを引き受けて、自社システムを自社で（その核心的な一部であれ）開発する能力を備えるようになるということは、避けて通れない途であるように思われる。その第一歩として、少なくとも自社の経営やビジネスプロセスにおける手段としての IT の有効性を見極め、事業としての競争力、IT 技術としての競争力を鑑みて、適切なシステムの調達形態を選択できるようになっておくべきである。そのための考え方を 4.7 節でも解説する。

1.4. DX の実践を支える人材

DX の取り組みにおいて、真に求められている「デジタル人材」には、もちろんプログラミングができたり、技術に精通したりしていることは重要な要素ではあるが、単にそれだけではなく、

事業や組織を深く理解し、そこにデジタルを組み合わせでどのような未来を描くのかを共有し、業務現場の人々と対話・議論ができる能力が求められる。これは一般に技術と事業の 2 つの側面の意味で「両利き人材」と呼ばれることがあるが、中核となって推進していく人材にはそれに加えて、事業推進者として経営層と対話する能力が必要となる。つまり経営層と対峙、対話ができる（事業と技術を組み合わせた結果の効果や経営的インパクト、予算管理等を説明できる）という意味で、経営という 3 つ目の側面を持ち合わせる人材も必要である。

特に DX 先進事例では、こうした事業・技術・経営の 3 つの観点に通じ、リーダーシップを発揮できる「ヤタガラス（八咫鳥）人材」が中心となり DX の方向性や開発推進、事業適用を牽引していた。経営の言葉で経営者を説得し、事業の言葉で事業部門を巻き込み、技術の言葉で開発メンバーと実現可能性の議論ができる。そのような人材が中心にすることで、スムーズに DX プロジェクトを立案・推進できる。しかし、ヤタガラス人材はどの組織にもいるわけではない。そのような組織では、内製力を強化するために、DX 先行事例では外部から双方に精通した人材を採用するような事例も少なくないが、前述したようにそもそもの人材が少ない事情もある。内部で育成を図るために、組織内で人事ローテーション制度などを活用し、事業の現場の人材をあえて DX プロジェクトに巻き込み、デジタル技術の知見や経営との接点を持ちながら、そうしたスキルセットを意図的に身に付けさせるような取り組みも見られる。また、一人で全てを背負い込むのではなく、チーム内で役割を分担して 3 つの役割を果たしているような事例も見られる。しかしながら、いずれにせよ、そうした DX を支える中核的存在が必要だとはいえそうである。



図 1.4.1 ヤタガラス人材のイメージ

DX の取り組みのように、事業や組織全体に変革を促すプロジェクトを推進するためには、事業・技術双方の視点をもって経営層と対話を繰り返し、まず、組織の構成員が腹落ちする必要がある。組織が目指すべき世界に向けて、デジタル技術を前提とした事業変革を継続するには大きなコストがかかり、組織で一丸となった協力体制が必要となる。例えば、慣れ親しんだ業務フローや製品が変わり、現場からの反発が予想されても、将来のために今必要なコストであると現場

の従業員が納得し、または割り切ってもら場合などである。これは経営層が掲げるビジョンの現場への浸透度や、その熱意、社内文化などにより左右されるため、なぜ自社がそのデジタル化への戦略をとるのか、自社がその技術を活用してどのような未来を目指すのかといった、事業・技術双方の視点での未来志向の対話を繰り返し、最終的には社員が腹落ちしていることが重要である。

こう説明すると DX の現場を支える鍵は、マネジメント能力、エバンジェリスト能力であると結論付けてしまいそうだが、少なくともその中に、デジタル技術への理解と DX に必要なソフトウェアを開発する内発的機能の強化も含まれていることは忘れてはならない。経営・事業・技術を繋ぎ合わせるヤタガラス人材の存在は重要であるが、前述した内製化できる組織成熟度を高めていくための人材育成・登用も怠るべきではない。むしろ、ヤタガラス人材が中心となって、内製化の重要性を訴え、また事業や経営に関わる人材に技術を伝播させるような取り組みを推進すべきである。

1.5. アジャイルマインド

DX の実践を支えるには、前項で述べた「ヤタガラス人材」が重要となるだけでなく、DX の取り組みに関わる人たちが、組織として対応を進めることが必要となる。

そうした中で、組織に属するそれぞれの人たちが、業務や組織などの変革を実現するためには、従来とは異なるマインドをベースとして持って、取り組みを進めることが必須である。

そうしたマインドに当たるものとして、「アジャイルマインド」という考え方が、IPA 内の人材育成を推進するコンテンツ¹³の中で紹介されており、本項では、その考え方について解説する。

そこで紹介されている、「アジャイルマインド」とは、

従来のように、問題を分析して最初にゴールを決め、そのゴールを固定して、そこに向けて論理的に解決していくという実践の仕方ではなく、

継続的な価値の実現のために、「現状を観察して」「失敗を恐れず」「問題を見つけ」「それに対して、仮説を立て」「関係者と協力しながら」「実際に試して」「試した結果、得られた結果を検証して」「それを元に問題や仮説の見直しを行う」ことを実践できるマインドとしている。

では、なぜ変化に対応していくために、既存のマインドを捨て、こうしたマインドを持つことが必要だろうか。

価値を生み出すにあたって、業務の内容自体が、相対的に不確実性が高い業務へシフトしてきているためである。

従来からあるルーチンワークなどの業務は、IT 化などにより、減っていき、よりイノベーティブな仕事が増えてきているというのが、大きな流れといえるだろう。

また、変化が激しいために、業務のゴール自体が変わってくるのでそれに適応するだけでなく、どのような課題をどのようなサービスやソリューションで解決していくのがよいのか、仮説を

¹³ <https://www.ipa.go.jp/files/000089485.zip> 内の「05_インプットトーク.pdf」

立てて、課題自体も自分たちで設定し直しながら、軌道修正を重ねつつアプローチしていくことが重要となってきたという点も挙げられる。

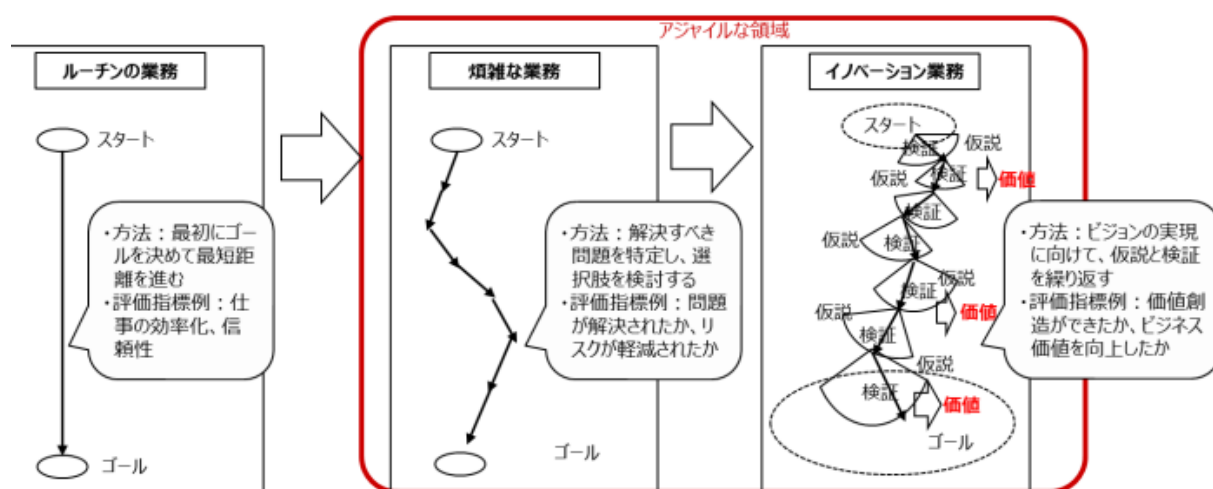


図 1.5.1 探索と適応による価値創造のアプローチ

こうしたアプローチをとりながら、ビジネス上の価値を実現するためには、不確実な問題に対して、探索と適応を繰り返しながら、解決していく必要がある。

つまり、自分たちのビジョンのもと、社会や顧客にとって、必要と考えられるプロダクトやサービスの仮説を立てて、それに基づいて設計開発を行い、社会や顧客からフィードバックをもらって、プロダクトやサービスの改良につなげていく、こうしたサイクルを継続的に繰り返すことが有効となる。

プロダクトやサービスの企画と設計開発を対にして、それらを進化させていく、こうした活動により、社会や顧客へ価値を届けることへつながると考える。



図 1.5.2 価値駆動とアジャイルな活動

そうした、業務を進めていくために、日々の仕事を進めていく上では、次にあげるアジャイルな仕事の原理を実践していくことが重要ではないだろうか。

- ・顧客と向き合うチームをつくる（顧客志向、顧客との協創）
- ・ビジョンとして目標を共有する（価値の共有、未来を創る）
- ・目標としてやるべき仕事について、優先順位を付けて、ひと仕事に小分けにして、毎週見直す（価値駆動、仮説検証型）
- ・ひと仕事は必ず対面で（ペアで）実施し、困り事はチームで解決し、仕事の結果は必ず現地・現物で確認する（人間中心、現地・現場・現物指向）
- ・毎週チームで、やるべき仕事を決める（自分事化）
- ・毎日チームで、やった事、やる事、困り事を共有する（自己組織化チーム）
- ・毎週、成果に対して何らかの形で顧客からのフィードバックをもらう（高速な仮説・検証サイクル、ようこそ失敗）
- ・毎週、チームで振り返りを行なって、仕事の進め方を改善し続ける（進化する組織、学習する組織）

そして、以上のような点を原理とし、チームで不確実性の高い問題の解決に取り組むことで、メンバー全体で同じ問題に取り組む、メンバーが持つそれぞれの得意分野を活かしながら、チームとして対応を進めていくことになる。その中では、自分の行った対応に対して、適宜フィードバックが行われたり、自分が持っていないスキルや技術を他のメンバーから学ぶ機会を得ることができたりする。そのようにして、チーム自体が学習する組織へとつながっていくことになり、学びの環境がチーム内に作られることで、チーム自体の進化を生むことにもなる。

このように、「アジャイルマインド」を持って、仮説と検証を繰り返しながら、価値を実現するための活動を進めていくことで、各チームが学習する組織となり、組織の進化へつながる。それとともに、組織運営の仕方も変わってくることとなり、組織自体の変革への下地にもなると考える。

第2章 DX を継続的に進めるための考え方

第1章までは主にDXを推進するための考え方を示してきたが、本章では継続的に進めるための具体的な対応を考えたい。

そもそも、企業によってそれぞれの生業や置かれている状況、またありたい姿や目標・ビジョンなどによって、選択できる対応は多様であり、よって課題やその解決方法も異なってくる。従って、DXの推進のあり方も多様である。また、本書では、「DX」について、既存事業の枠組み内での変革を表す「オプティマイゼーション」と、既存事業の枠組みを超えた変革を表す「トランスフォーメーション」の両方を、多くの企業がそれを一連のものとして捉えていることから共に含めて考えているが、これもDXの具体的なイメージを複雑化させる。しかし、こうやってしまうと、ではDXを推進するにあたり、企業として如何に対応すべきか、単に困惑を惹起するだけだろうと思う。

我々も、先進事例を調査する中で、このような複雑さに直面した。しかし、2021年度の調査では企業として実現していくトランスフォーメーションと、部門の中からスモールスタートを切ることができるオプティマイゼーションで、DXの取り組み方において重視するポイントが違うという気づきがあった。そこで、トランスフォーメーション中心かオプティマイゼーションを中心とするかによって生まれる変革の規模感の違いに注目して、この取り組み方を整理できるのではないかと、という仮説を立てこれを検証するように、過去例の整理を再度試みることにした。

この整理をするにあたり、変革の規模感を「変革規模」という指標に落とし込んだ。この指標は様々な企業や組織や事業のDXを推進する中で、組織の内外を含め変革や効果の波及範囲の大きさを示すものとして、既存事業における社内を中心とした変革を目指すオプティマイゼーション(3段階)と、新商品や新サービスを通して、市場や社会の変革を目指すトランスフォーメーション(4段階)を合わせ、7段階を設定した。

そして、取り組み方については、経営や組織といった体制、IT人材や技術力など、DXを実現するために要求される要素群として整理し、これを1～5の値をとる39の指標にまとめた。この指標群を「組織成熟度」と総称することにした。

この調査では、ヒアリングやアンケートによる先行事例調査の結果を、変革規模によって組織成熟度がどのように異なるかを整理することで、多様な企業におけるDXを進めるための考え方及び、取り組み方の共通点の発見を試みており、本章ではその結論の要点を紹介する。

より詳細な内容については、「DXの継続的な取り組み事例に関する調査 概要報告書」¹⁴にも記載しているので、是非そちらも参照していただきたい。

¹⁴ 独立行政法人情報処理推進機構 「DXの継続的な取り組み事例に関する調査 概要報告書」 2022年4月
<https://www.ipa.go.jp/files/000097139.pdf>

2. 1. 変革規模と組織成熟度の定義

2. 1. 1. 変革規模の考え方

冒頭でも触れたが、変革規模は、既存事業内での変革を目指すオペティマイゼーションと既存事業の枠組みを超えた変革を表すトランスフォーメーションに大きく分けられる。これらの変革において、オペティマイゼーションの段階には、部門内や取引先間での業務変革を、トランスフォーメーションの段階では、新商品やサービスを通じた市場や社会の変革等といった取り組みを含んでいる。これらのオペティマイゼーションとトランスフォーメーションは並行に行われる場合もあるが、今回の調査で用いるレベルでは、形式的にオペティマイゼーションから行う場合を想定してレベルの設定を行った。

変革規模のレベルの設定にあたっては、「顧客体験を向上させているか」、「既存業務を効率化させているか」、などといった様々な要素を考慮したうえで、オペティマイゼーションについて3レベル、トランスフォーメーションについて4レベルを設定する。概要報告書では、変革規模のレベル判定ツリーも記載しているので、詳細な内容に関しては、そちらも参照していただきたい。以下では、今回設定した7つのレベルの具体的な中身について示していく。

(デジタルオペティマイゼーション)

(1) 一部の業務変革

ある特定部門の業務を対象とし、DXの入り口としてRPA（Robotic Process Automation）やWeb会議の導入を行うことで、主に売上原価削減や販管費削減を実現している。

(2)-a 企業全体の業務変革

組織（企業）全体の業務を対象とし、複数業務・部門を跨った取組みを行うことで、生産性改善、既存売上改善、生産設備等の最適化、財務の最適化等を実現している。

(2)-b 取引先も含めた業務変革

取引先も含めた組織全体の業務を対象とし、企業内に限らず取引先を含めたサプライチェーンを改革変革することで、生産性改善、既存売上改善、生産設備等の最適化、財務の最適化等を実現している。

(デジタルトランスフォーメーション)

(3) 顧客体験の変革

取引先・顧客など当該企業外のプレイヤーの体験を変革することで、新たな提供価値を生み出し、顧客体験高度化、在庫と収益のバランス最適化等を実現している。

(4) 市場での競争力の変革

生み出した提供価値の向上により競争力を高めることで、当該企業の市場での競争力の変革を実現している。

(5) 市場での立ち位置の変革

新収益源/サービス・製品の創出、成果分配型事業展開などによって、業界内の構造や市場その

ものの変革を実現している。

(6) 社会の変革

隣接市場・新規事業展開、プラットフォーム・ビジネス展開などによって、消費者行動、業界内外の構造、労働構造など社会の行動や構造の変革を実現している。

		定義			典型例	“一言”で言うと・・・	
		範囲	考え方	深さ(DX種類＝得られる成果)			
DX推進の取り組みの範囲	デジタルトランスフォーメーション	(6) 社会の変革	社会	消費者行動、業界内外の構造、労働構造など社会の行動や構造が変革される	・ 隣接市場・新規事業展開 ・ プラットフォーム・ビジネス展開	・ UberEATS ・ Airbnb ・ インスタグラム	社会を変えた
		(5) 市場での立ち位置の変革 (により、リーダーに)	市場 (での自社の立ち位置)	業界内の構造や市場が変革される	・ 新収益源/サービス・製品の創出 ・ 成果分配型事業展開	・ KOMTRAX	市場を変えた (例:Pfizer®を1つ以上展開)
	事業変革	(4) 市場での競争力の変革	競争力 (事業成長力)	提供価値が向上する 価値向上により、当該企業の市場での競争力が変革される	・ 新収益源/サービス・製品の創出	・ 音楽をサブスクで聴く	競争力を変えた
		(3) 顧客体験の変革	顧客体験 (取引先等も含む)	新たな提供価値を生み出す 取引先・顧客など当該企業外のプレイヤーの体験が変革される	・ 顧客体験高度化 ・ 在庫と収益のバランス最適化	・ ネット音楽を聴く	売り物の価値を高めた
デジタル・オフィスマイゼーション		(2)-b 取引先も含めた業務変革	取引先も含めた組織全体の業務	企業内に限らず、取引先を含めたサプライチェーンが改革される	・ 外部(取引先)を巻き込んだサプライチェーンの改革	・ 取引先も含めたEnd-to-EndのSCMの自動化	サプライチェーンを変えた
		(2)-a 企業全体の業務変革	組織(企業)全体の業務	複数業務・部門を跨った取組み 個々の業務・部門改革が複数あるだけであれば(1)に該当	・ 生産性改善、既存売上改善 ・ 生産設備等の最適化 ・ 財務の最適化	・ 需要に基づくSCMの自動化(工場の自動化、物流、データ連携)	業務を変えた (企業業務の50%以上を変革)
		(1) 一部の業務変革	ある特定部門の業務	DXの入り口として、RPAやWeb会議の導入を行う	・ 売上原価削減 ・ 販管費削減	・ 自動発注(小売)	一部業務を変えた

図 2.1.1 変革規模一覧

2.1.2. DXの実現に必要な組織成熟度

変革規模と同様、DXの取り組みを整理するために、DXを実現していく上で必要と考えられる経営体制・環境準備・技術力などの個別要素を集約し、その達成度を組織成熟度として定義した。

組織成熟度を測る指標については、2021年度の調査を行った、「DX実現を可能にするITシステムを目指す実践手引書作成のためのDX実践事例調査」とDX推進指標をふまえて、指標の粒度・項目数の適正なレベルを考慮し、経営、事業、技術、人材・組織のテーマについて26の観点を設定し、それらの観点のレベルを測るために39の指標を整理した。これらの指標は、それぞれ、取り組みを始める前から、最適化を済ませるまでの5段階のレベルで計測することができる。

26の観点は、ヒアリング調査実施前にまずは仮説として整理を行い、ヒアリング調査後の整理・分析を経て、いずれもDXを実現していく上で必要な指標であることを確認した。

また、39の指標は、図2.1.2のように組織成熟度指標に対応する推進指標が存在する。

各指標、観点のレベル感の詳細は「DXの継続的な取り組み事例に関する調査 概要報告書」に記載されているので、そちらも参照していただきたい。

組織成熟度				DX推進指標	
#	観点	評価指標	評価項目	#	推進指標項目
1	経営の仕組み	ビジョンの立案・共有	ビジョンの提示、共有	定性指標No.1	ビジョンの共有
2		ロードマップの作成 (やるべきことの明確化)	ロードマップの作成	定性指標No.7-1	戦略とロードマップ
3		新しい開発手法の適用	アジリティ	定量指標No.22	アジャイルプロジェクトの数 [件]
4		資金、人材、技術の集約	DX予算管理	定性指標No.4-4	投資意思決定、予算配分
5			やるべきことの明確化	定性指標No.7-1	戦略とロードマップ
6		経営者のリーダーシップ (経営者の理解、実践と権限移譲)	理解	定性指標No.3	経営トップのコミットメント
7			実践	定性指標No.3	経営トップのコミットメント
8			権限移譲	定性指標No.5	推進・サポート体制
9		事業への落とし込み (PoCからの脱却)	PoCからの脱却	定性指標No.8	ビジョン実現の基盤としてのITシステム の構築
10		アイデアの創出の仕掛け	アイデアの創出	定性指標No.9-3	事業部門のオーナーシップ
11		評価指標、基準と仕組み	指標	定性指標No.8-4	IT資産の分析・評価
12			評価の仕組み	定性指標No.9-6	IT投資の評価
13			プロジェクト撤退基準	定性指標No.8-5	廃棄
14		データへの投資	データの収集	定性指標No.8-1	データ活用
15		セキュリティ・ガバナンス	IT・セキュリティガバナンス	定性指標No.9-5	プライバシー、データセキュリティ
16			DX重点配分	定性指標No.5	推進・サポート体制
17	事業	既存事業・業務の連携 (既存事業へのマイナス影響の受容)	既存業務との連携	定性指標No.7	事業への落とし込み
18			既存業務へのマイナス やDX失敗への構え	定性指標No.4-1	体制
19		社内の巻き込み	社内の巻き込み	定性指標No.6-2	技術を支える人材
20			セキュリティ方針	定性指標No.9-5	プライバシー、データセキュリティ
21			顧客接点データ	-	該当項目なし
22	技術	DX基盤の疎結合化・共通化	基盤	定性指標No.8-3	全社最適
23			システム連携	定性指標No.8-7	非競争領域の標準化・共通化
24		新技術の検討・取り込み	新技術の検討	定性指標No.6-2	技術を支える人材
25		レガシー刷新 データ価値向上、活用	計画・管理	定性指標No.8-8	ロードマップ
26			価値のあるデータ	定量指標No.19	データ鮮度 [リアルタイム/日次/週次/月次]
27		クラウド活用	クラウド活用	定性指標No.8-2	スピード・アジリティ
28		アプリ軽量化、スピード化	アプリの軽量化	定性指標No.8-2	スピード・アジリティ
29			システム	定性指標No.8-5	廃棄
30		データ収集、保管、 エンジニアリング	データの収集、活用	定性指標No.8-1	データ活用
31	人材・組織	DX組織整備	ITと事業の融合	定性指標No.5-1	推進体制
32			DX組織配置	定性指標No.5	推進・サポート体制
33		失敗からの学び	失敗からの学び	定性指標No.4	マインドセット、企業文化
34		八咫烏(やたがらす) 人材の整備	配置、採用	定性指標No.9-2	人材確保
35			育成	定性指標No.6	人材育成・確保
36		自社開発の内部エンジニアの整備	自社開発の内部エンジニア	定性指標No.9-2	人材確保
37		データ活用人材の連携	データ活用人材	定性指標No.9-4	データ活用の人材連携
38		開発・運用の内製化	開発・運用の内製化	定性指標No.9-2	人材確保
39		外部リソースの活用	外部リソース	定性指標No.5-2	外部との連携

図 2.1.2 組織成熟度と推進指標の対応表

2.2. DX を継続的に進めるための有効な施策

本節では、前節で定義した指標を用いて、先進企業へのヒアリング調査を整理した結果得られた、DX を進めるためのアクションを紹介する。DX を先進的に取り組んでいる企業に共通する点として、次の三つのアクションを行っていた。

- ① トップダウンとボトムアップの両面から取り組みを始める
- ② 組織内外に取り組みを波及させるための環境の整備を行う
- ③ デジタルカンパニーのスタイルやマインドを定着させる

本節では、これら三つのアクションについて、先進企業のヒアリングから得られたキーメッセージを、前項で定義した組織成熟度指標と対応付けながら、節毎に解説していく。

2.2.1. トップダウンとボトムアップの両面から火をつける

DX の取り組みを始めるにあたっては、社内全体で DX 推進における意識を高める必要がある。故に、トップダウンとボトムアップの両面から、DX に関するビジョンを発信し、意識の統一を図るといった考え方が有効だ。経営陣が強いコミットメントと強烈的なビジョンの発信を行うことが重要である一方で、課題意識の強い従業員を焚き付け、アイデアを拾い上げることも重要になってくるからである。この考え方を実現するためには、経営陣が社内外に DX 推進の明確なビジョンを提示したうえで、現場が手を上げやすい環境を築く必要があると言える。以下では、その様な環境を整えるために参考になるメッセージを、経営、事業、技術、人事・組織のカテゴリに分類し、組織成熟度指標の項目毎に、DX 先進企業へのヒアリングを通して得られたキーメッセージと概要を記していく。

【経営】

経営陣はトップダウンだけではなく、ボトムアップを可能にする環境を整える必要がある。同時に、各々で吸い上げた内容を DX のビジョンやロードマップに反映して展開しなければ、DX を進めることができない。

以下では、組織成熟度指標である“経営者のリーダーシップ”にてトップダウンの考え方、“アイデア創出の仕掛け”にてボトムアップの考え方を示したうえで、両者を結ぶようなアクションを“ビジョンの立案・共有”、“ロードマップの作成”で示す形で、キーメッセージをまとめた。

■経営者のリーダーシップ

- 経営者自身が強い危機感を持ち、その覚悟を示せ

- 経営陣はデジタル技術の特性を理解し、それが経営へどのように影響するかを意識して、意思決定せよ
- 変革を実施する事業責任者をおき、目的、期待成果を明確にしたうえで、権限移譲しろ

リーダーシップは、経営者自身が DX を深くリードすることだけでない。企業の現状把握や、計画を経営者が行ったうえで、実行フェーズにて影響力がある人物に適切な権限移譲を行うことで、DX を加速させることもリーダーシップの要素である。

■アイデア創出の仕掛け

- 社内外の声を収集する仕組みを設けよ。アイデアは前提をつけて抑圧・抑制をしない。まずは現場に自由に発想させよ
- 世の中に影響力のあるトップユーザーと協力しイノベーションパイプラインを構築しろ
- 今までの延長線上ではないアイデアを生み出すための人材と組織を考えろ

現場の課題を解決する際に、アイデア創出の仕掛けは非常に重要なポイントであり、多くの企業で DX に繋がるアイデア創出の仕掛けを設けている。ユーザーの声に耳を傾けるのは勿論のこと、トランスフォーメーションを行う際には、今までの延長線上にない組織や、仕組みを活用することも有効である。例えば、世の中に影響力のあるベンチャー企業との連携や、産学連携を行うことで、先進的なイノベーションに繋がることもあると言えるだろう。

■ビジョンの立案・共有

- 各事業部の課題を熟知している精鋭メンバーを集め、対話を繰り返すことで DX の戦略を立案せよ
- ビジョン策定は大きな戦略を Push し、戦略に従った現場の具体戦略を Pull することで整合性と実現性を両立せよ

上から押しつけられた、現場で現実感のないビジョンでは実現されないし、逆に現場に寄り添いすぎたハードルの低いビジョンであれば、変革の成果を得ることが難しい。そのため、経営者と現場の事業を熟知しているメンバーとの対話等を通して、このバランスをとったビジョンを策定する必要がある。

■ロードマップの作成

- デジタル技術を活用して短期的なビジネス目標（生産性向上、顧客獲得、利益率アップ等）を達成することで人と資金および変革の勢いをつけ、デジタルを活用した新領域のビジネス開拓を進めろ
- RPAなどの個別業務の効率化技術はDXの手段であって目的ではない。これらの効率化技術によってデジタル化したデータや省力化によって得られたリソースを使い次の変革を目指せ
- リソースを無駄にしないために何をやらないかを決めろ
- 社会変革を誰もが目指すべきとは限らない。市場や自社の立ち位置を意識してやるべきことを考えろ
- 一度決めた長期的なロードマップに固執するな。目標と達成時期を明確化したうえで、定期的に状況によって方向修正しろ

DXを推進する企業がロードマップの作成を行う際は、現場の課題を熟知しているメンバーとの対話をしたうえで、明確なロードマップを策定することが好ましい。実際に、ヒアリングした先進企業でも、高い変革規模を実現している企業ほどビジョンをロードマップに落とし込み、やるべきことを明確化している傾向が見られた。ここで注意していただきたいのが、描いたロードマップに固執してはいけないという点だ。DXに関する取り組みは、従来の取り組みに対して不確定要素が多く、障壁や課題も生じやすい。そのため、状況に応じた柔軟な修正がより求められる。大切なのは、一度描いたロードマップに沿ったアプローチを愚直に行うことなく、目指すべきビジョンをいかに実現するかという考え方である。

【事業】

事業カテゴリでは、組織成熟度の”社内の巻き込み”に焦点を当てて、ボトムアップに関するキーメッセージを紹介する。“社内の巻き込み”という指標のヒアリングでは、DXの先進企業の各社が苦労し、様々な工夫を行っていたという声を聴くことができた。多くの先進企業がつまずいたポイントとして、この指標は重要性が高いと思われるので、是非、DXに取り組む際は参考にしていいただきたい。

■社内の巻き込み

- 従業員からボトムアップに意見をもらうためには、経営がその意見を受け止めているというフィードバックをして、意見を言うことに対する安心感を与えろ

- 問題意識があり、やる気のある従業員が手を上げやすいような環境を整備しろ
- DX の取り組みの当初は、PoC の助成金、チャレンジを推奨する人事制度など、経営側からのサポート策を打ち出し、DX に積極的に取り組もうという機運を高めよ
- 事業部に解決策を提案する際には期待値コントロールが重要。予算・体制上すべてにおいて 100 点は望めないことを共有して、課題を解決できる落としどころを示せ
- 外部の媒体の取材を受けることで、社内のプレゼンスを上げろ

社内を巻き込む際には、従業員が意欲的になるような空気づくりが重要である。ビジョンを示して意見をもらったものの、自分に都合の悪い部分で意思決定を先送りにしたり、通常業務では今まで通りのやり方を続けていたりすると、従業員の安心感はすぐに不信感となり、正直な意見が出なくなり、ビジョンも形骸化してしまう。そうならないためには、例えば、現場から拾った声を全社に共有することで意見を発信することに対する安心感を作り出すことや、公募やチャレンジ推奨といったやる気のある従業員を焚き付けるサポート策等の取り組みが有効である。また、企業によっては、外部メディアの記事に取り上げられることで、それが社内に還流し課題意識が社内に広がるというように、社外のマーケティングを活用して社内への課題意識を高める施策も見られた。

【人材・組織】

トップダウン、ボトムアップ、それぞれのビジョンを結びつける企業組織面からの対応を図るのが”DX 組織整備”の項目である。ここで組織整備というと、ただ IT 人材を集めた組織を創設すればいいと思われるかもしれないが、それだけでは十分ではない。この組織は企業の IT システムに精通するのみならず、既存の事業的枠組みや現場運営と連携が取れていなければ、効果的に DX を進めることは難しいのである。

■DX 組織整備

- IT 部門や DX 推進部門はビジネス部門にも席を作り、ビジネス課題とデジタル技術を繋げる通訳となれ
- DX には必ず事業の現場を巻き込んでビジネスとしてのゴールを明確にして進めろ

DX の組織配備に関して、オプティマイゼーションとトランスフォーメーションに共通して言えることは、DX 推進組織とビジネス部門が近い距離で連携をとることが重要な要素であるということだ。どのような変革をするかだけでなく、どう変革をするかが、現場組織の十分な理解と協力の下、行われないと無駄や混乱を招くからである。ただし、連携の取り方は様々で、各ビジネス部門のキーマンを DX 推進組織に配置するだけでなく、逆に IT 側の社員をビジネス部門に送るといった施策も、また有効的である。

2.2.2. 活動を全社に波及させるための身体作りを行う

前節に見られるような取り組みで DX 推進に対する意識が向上した後は、活動を全社に波及させるための仕組みやプロセスの整備を行う必要がある。DX は、DX 推進部門だけが頑張ってもうまくいかない。そのため、全社的に部門を跨るように広める必要がある。このように、活動を全社に普及させるための取り組みが必要になってくる。具体的な内容としては、ビジョンを実現する上で必要な人材の育成・獲得や、アイデアの実現を加速させる環境を整備することなどが挙げられる。また、全社への波及だけでなく、DX の活動を減速させないためのプロセスも整備しなければいけない。例えば、DX 事業の失敗に対する向き合い方や、評価制度を設けなければ、活動が停滞してしまう恐れもあるだろう。本節では、この様な活動を波及させるためのキーマッセージを事業,経営,人材・組織,技術の 4 カテゴリー毎に整理した。

【事業】

事業に関しては、2 つの組織成熟度指標に関するキーマッセージを挙げる。活動を全社に波及させるにあたって、取り組む事業の検証や、既存業務との関わりは重要である。変革規模の「(1) 一部の業務変革」や「(2)-a 企業全体の業務変革」で特に見られるような、現状の事業を発展させる形の DX では、現場への配慮を怠ってしまうと、その活動が徒労になりかねない。逆に、トランスフォーメーションの様に、新しいサービスを生み出すような DX では、現場への配慮が DX 遂行の妨げとなってしまうこともある。故に、いずれの例においても、事業への取り組み方次第では、DX の推進を減速させてしまう危険性がある。よって、いずれのタイプの DX を推進していくにせよ、DX を減速させずに推進し続けるためには、“既存事業・業務の連携”, “事業への落とし込み”といった点を考慮せざるを得ない。以下では、これら 2 つの組織成熟度指標と関連付した視点で、DX の取り組みを滞りなく全社に波及させるために有効なプロセスや仕組みを記載する。

■既存事業・業務の連携

- DX が現行事業に悪影響を与えると懸念する人には、長期的・経営的な視野でのあるべき姿を丁寧に説明し、粘り強く説得せよ
- 自社のブランドがエコシステム作りの障害になるのであれば、別会社・別ブランドを立ち上げろ

自社の差別化に繋がっているデジタルサービスを、同業他社に外販し、収益を拡大させるようなケースでは、現行事業からの不安・抵抗がある場合もあるだろう。それでも進める価値がある場合は、その価値を丁寧に粘り強く説明することや、既存ビジネスの妨げにならないように別会社化するという対応をとることができる。

特にトランスフォーメーションにおいては、事業のカニバリズムを起こすなど、既存事業にマイナス影響を与える状況もあるため、不安や抵抗を払拭する取り組みは必要となってくると言えるだろう。

■事業への落とし込み

- 0 から 1 を作るためには、多くのアイデアを出し、それを早く安く作って、市場で検証しろ
- 何を Proof するかを明確にし、PoC のための無駄な PoC を行わない。プランニングがしっかりしていれば必ずしも PoC を行わなくてもよい

事業に落とし込む際には、行っている活動の検証が非常に重要になってくる。多くのアイデアを出し、仮説検証を素早く繰り返す考え方も重要であるが、一意一概にそれが全てとは言いきれない。ユーザーに必要最小限の機能を提供できるプロダクトのことを MVP と呼ぶが、闇雲に PoC をするよりも、自信を持って進められる場合は、MVP のようなミニマムなプロダクトを実用できるものを作るためのプロセスを取ることが大事という考え方も聞くことができた。この様なプロセスを繰り返して成功体験を積み上げることで、社内への波及効果は高まっていく。どちらにも共通して言えるのは、事業へ落とし込む際はスピード感が肝となってくるといえることだろう。

【経営】

経営という視点では、3つの指標に関するキーメッセージを記載する。具体的には、DX 推進を減速させないための仕組みとプロセスとして“評価指標、基準と仕組み”を、DX のビジョンを実現する上で必要な人材獲得という視点で“資金、人材、技術の集約”を、環境整備を進める上での観点として“セキュリティ・ガバナンス”といった3つの指標を挙げた。いずれの指標も、DX の活動を全社に普及するにあたって有効なアクションなので、今一度意識していただきたい。特に“セキュリティ・ガバナンス”で注意していただきたいのは、闇雲にセキュリティを強化すれば良いわけではないという点である。セキュリティを必要以上に強化することで、むしろ DX の推進を阻害する可能性もあるということは、理解いただきたい。

■評価指標、基準と仕組み

- DX の成果は、従業員や現場にとって分かりやすい指標を使い
- 自社の DX の推進状況を定期的に客観的な指標で把握し、行った施策と合わせてノウハウの蓄積や施策の見直しにつなげろ
- 個々の案件評価については明確な指標を設けず、都度経営層で議論せよ

DX の評価指標に関しては、それまでの企業の取り組み方に応じて、一定の指標を設定するパターンと、明確な指標を設けないパターンに二分された。既存事業の延長線上にあるような取り組みを行う場合は、現場に伝わりやすい指標や、DX 推進指標を用いて客観的な指標で見ることを意識している。一方で、プロダクトによって KPI (Key

Performance Indicator) が変わるような DX 事業を行っている場合は、一定の指標を設けず、都度経営層などで議論することに重きを置いていた。どちらの場合も、自社の状況や DX の目的に応じた対応が求められると言えるだろう。

■資金、人材、技術の集約

- DX 推進チームに権限と資金を与え、他の組織にとらわれずに自由に動ける環境を与えよ
- どう社員を動かしたいかのインセンティブを意識して予算制度をデザインせよ

新しいことに取り組む場合、従来のプロセスに従うと時間がかかりすぎるため、DX 推進チームに権限と資金を与えて、PoC の助成金制度を設けるなどして、自由に迅速に動ける環境を構築している企業があった。また、予算制度の作り方から部門間の交流が生まれ、アイデアの拡大に貢献するケースもある。

■セキュリティ・ガバナンス

- ルールや原則として大方針を示しつつ、それぞれのユースケースについてケースバイケースで判断せよ

DX 推進においては、凝り固まった考えが阻害要因となりうる。よって、セキュリティ・ガバナンスにおいても、大方針・大原則を守りつつ個別のケースについて柔軟に対応できることが求められている。

【人材・組織】

人材・組織については、それぞれの視点から特徴的な 2 つの組織成熟度指標をピックアップした。人材面では、組織マネジメントの中核を担う”八咫鳥（ヤタガラス）人材の整備”であり、組織面では継続性を支える“失敗からの学び”に関する心構えである。重要なことは、人材と組織のどちらか一方が大事なのではなく、それが相互に影響し合って、DX のパフォーマンスを上げていくということである。

■八咫鳥（ヤタガラス）人材の整備

- デジタル人材の育成はキャリアパスと紐づけよ
- 業務を知っているパッションのある人材に IT 技術の教育を受けさせて、両方の会話のできる人間を育てろ

今回調査した多くの先進企業で共通して、レベルはさておき、全社員がデジタル技術を活用できる人材である必要があると考えられている。また、全社的な活動として全社目標を設定し人材部門と連携し、キャリアパスとも紐づけながら進められている。また、全社的な活動と別に、業務と技術の両方の会話ができる八咫鳥人材を育てようとしているケースもあり、社内から八咫鳥人材を育成する流れが見られた。

■失敗からの学び

- どのように早く失敗するかが重要。官僚的な構造を回避しスピード感を保て
- 失敗を分析するな。それよりもいかに早く失敗から立ち直るかを考えろ。立ち直りさえすれば、失敗したことは誰も気にしない

日本企業においては、失敗プロジェクトの分析に関して、プロセス整備や仕組み化を行わずに非効率的に行われることが多い。一方で、海外先進企業からは、失敗を分析し、学びを得ることよりも、失敗したことは認識した上で、スピーディに次の取り組みに移ることが重要との声もあった。例えば、とある海外企業では、迅速に行動できる小さなチームを立ち上げて、実験と学習を繰り返すために早く失敗することを重視している。また、早めの失敗を繰り返すことにより取り組みにおける仮説の誤りに気づき、その誤った仮説の軌道修正を行うことで、誤った仮説の延長でビジネスを考える無駄を回避することができる。このような失敗に対する分析に捉われない備え方や、考え方は日本企業にとっても学ばべき点が多いかもしれない。

【技術】

DX に関する取り組みを行う際、その際に社内で使われている技術に関して見つめ直すタイミングがあるだろう。その際、“レガシー刷新”は DX を効果的に実現するために、避けては通れない視点の一つである。勿論、レガシー刷新は一朝一夕で実現できることではないので、優先順位を決めながら取り組みつつ、並行して、データ活用やマイクロサービスを行うための基盤である、DX 基盤の疎結合化・共通化を進めることもしばしば行われる手法である。また、“クラウド活用”は、これらのシステム整備において今や不可避な手法である。だが、それに留まらず、顧客にも、クラウドの特性であるリスクの分散によって、安定環境の稼働の提供等といった意味で影響する。

以下では、DX の社内波及の際に重要となる技術という面から、前述の 3 つの組織成熟度指標になぞらえたキーメッセージをまとめていく。

■ DX 基盤の疎結合化・共通化

- DX 基盤の導入とシステム刷新は並行して進めよ。両者が関係する個所の刷新を優先的に進めるべし

- 基盤構築は目的ではなく手段。顧客に何を提供するかを明確にし、そのための基盤がどうあるべきかを考えろ

DX 基盤の疎結合化・共通化を考えたとき、当然ながら、何を実現したいか、何のために基盤を構築するかによって必要となる基盤の姿は異なってくる。

その際に共通して言えるのは、システム全体最適まで及ばなくても、疎結合な DX 基盤が構築されていればよく、レガシーが残っていても DX は進められるということだろう。

実際に今回の調査において、多くの企業が完全なシステム刷新を待たずして、疎結合な DX 基盤の構築や、DX の各種施策をシステム刷新と並行して実行していた。疎結合化は、DX を推進する上では、ビジネスドリブンで利用テクノロジーが交換可能となるような「コンポーザブルなアーキテクチャ」にもつながり、新しいよりよい技術を導入しやすくする効果も期待できる。

■レガシー刷新

- 闇雲にレガシー刷新するな。刷新のインパクトや市場変化の状況を考え、優先的に刷新すべき機能を見極めろ
- システム刷新は IT の問題ではなく業務の問題。事業部が中心となり、グローバル標準の業務スタイルへの変革を進めよ

DX 先進企業にヒアリングした結果は、レガシーを完全に刷新している企業は一部で、多くの企業が古くてリプレースしないと困るもの（実際には、困る手前の、業務継続リスクが許容範囲を超える時点で刷新すべきだが）やビジネスメリットが大きいものなど、戦略的意図に基づき必要な範囲についてレガシーを刷新していた。前述のように、必ずしもレガシーの完全刷新は、DX 先進企業でも求められないだろう。一方で、グローバル全体で業務プロセスやデータを標準化するビジネス変革プロジェクトとして、大規模なレガシー刷新に取り組む企業もある。この場合、プロセスオーナーといった形でビジネス側にオーナーシップを持たせて進めることがカギとなる。

■クラウド活用

- 小さく始めたサービスが、大きく成長しても運用で困らないように、初めからスケールアウトを意識した作りにせよ
- ガイドライン・コンセプトを策定することでクラウド活用を促進せよ

本書を記載するにあたって調査した企業では、ほぼすべての企業が積極的なクラウド活用を行っていた。その際に、クラウドの方針やガイドラインをしっかりと定める必要がある。また、新しいサービスを展開する上では、顧客が大きく増えた場合でも容易にスケー

ルアウトできるようにしておくなど、クラウドの強みを活かすことが重要となってくる。方針やガイドラインの策定に際しては、4.1.5 節の「あるべき IT システムとクラウドの関連」解説も参照していただきたい。

2.2.3. デジタルカンパニーの流儀を体得する

本節では変革規模をより高め、トランスフォーメーションを行う際に重要となるキーメッセージを紹介していく。ここでのデジタルカンパニーとは、DX レポート 2.0 で定義されている、デジタル技術を駆使して価値を創造するデジタル企業のことを表している。変革規模を更に高め、デジタルカンパニーを目指すにあたっては、データ活用や、新ビジネスや新市場を共創する企業間でのエコシステムの形成など、より高度な取り組みが求められる。また、高いアジリティを実現するためのスキル、メソッド、マインドを取り入れることも重要になってくる。これらの取り組みに関するキーメッセージを、本節では経営、事業、人材組織、技術といった観点で紹介していく。

【経営】

経営では、“データへの投資”が特に重要なポイントである。DX を加速させるためには、社内で取得しているデータだけでは不十分で、外部パートナーと連携することで必要なデータを取得し、質の高いソリューションを展開していく方法も、効果的である。

■データへの投資

- 目指す姿を踏まえ、直接取得するデータだけでは補えないデータを外部から入手せよ

企業がデジタルカンパニーに生まれ変わっていくために欠かせないのがデータである。なぜならば、経営の判断の中心がますますデータとなってきたからであり、つまりデータに裏付けられた根拠をもってデータドリブン経営をしていかなければならないからである。しかしながら企業や組織が収集できるデータには限界がある。時にはビジネスに必要なデータは、外部から交換や購入といった方法で入手しなければならなくなっている。

- DX 先進企業からは、DX を加速させていくためには、社内で取得しているデータだけでは不十分で、外部パートナーと連携することで必要なデータを取得し、質の高いソリューションを展開していくという声が聞かれた。

金融業・保険業 AH 社では、顧客企業が保有しているデータに、自社で保有しているデータを組み合わせることで新たな価値を提供している。例えば、化学工場の顧客の場合、事故の予兆を検知し、事故を減らしたいといったニーズがある。その事故の予兆を知るためには、顧客企業が持っているデータに、自社が持っている化学工場に関わる過去の事故データを組み合わせることで、精度の高い予兆検知を実現している。

【事業】

事業面では、“データ価値向上、活用”といった組織成熟度指標が大きく関わってくる。データの活用は DX のベースとなる活動である。トランスフォーメーションに限らず、オプティマイゼーションの段階まで幅広くデータ価値向上、活用が行われている。今回調査したいいずれの企業も、集めるべきデータを見極め、データを連携させることでさらなる活用につなげている。

■データ価値向上、活用

- データは集めるだけでは意味がない。ビジネスで使える形にするために、集めるべきデータを見極め、各所から収集したデータを連携できるようにしろ
- データ利活用の際には、お客様を集団としてではなく、ひとりひとりとして向き合え

事業において、より多くの情報を収集し将来へのビジネスの精度を上げていく必要がある。現在は、データの収集により、過去をより正確に知ることはできるようになった。しかし、それだけでは精度の高い計画を立てることができるとは限らない。必要なデータが何か、またどのように収集すべきか、不足するデータは何か、補足する方法はないかといったデータそのものに対する配慮も必要となり、同時にそうしたデータ群を連携し分析してはじめて市場や個人と向き合うことができるわけである。

- データの活用は DX のベースとなる活動。オプティマイゼーションの段階からトランスフォーメーションの段階まで幅広くデータ価値向上、活用が行われている。AI による来客予測(店先のカメラセンサー、天気予報、訪問者数等のデータ)を元に食材の準備や従業員配置に活かしている企業、顧客軸にデータを集めるために発生したデータをカードシステムに連携させている企業、保有する 2-30 のサービスに対して 1 つのマスターアカウントを作成して管理することで顧客の UI を高めるとともに顧客データを一元化している企業など。いずれの企業においても集めるべきデータを見極め、データを連携させることで更なる活用につなげている。

【人材・組織】

デジタル企業目指すにあたっての最大の課題は人材・組織である。これはデータを活用して事業変革を導ける人材の存在がデータ活用を行う大前提となるからで、これが事業そのものと深く関わることに鑑みれば、この人材を外部に任せていてはいけないことは言うまでもないだろう。同時に、このデータ活用を支えるシステム整備やデータそのものの取得の仕組みを構築することも重要となる。こうした諸点に対応する人材・組織に関する 5 つの組織成熟度指標について、キーメッセージをまとめていく。

■データ収集、保管、エンジニアリング

- 基幹システムのデータを全社で統合し、経営判断するデータを得ろ
- 消費者と直接つながり、情報を獲得せよ

基幹システム刷新の観点の一つとして、業務プロセス・データを標準化することで、地域や会社によって異なるデータを揃えることができる環境を構築する、という視点がある。これはあくまで一つの例にすぎないが、データの統一を図ることは、迅速な経営判断に直結する。また、データ収集といった観点では、消費者中心に捉えて、ユーザーの生の声を収集することで、顧客体験の変革を起こすこともできるだろう。

■データ活用人材の連携

- 業務が分かるデータサイエンティストを計画的に育成せよ

データ収集を行っても、データを経営判断や事業の推進に活かせなかったら、データを収集したことによって受ける恩恵は大きくはないだろう。しかしながら、業務も分かってデータ分析やデジタル活用もできる人材は少ないため、全社レベルで計画的に育成していくことが好ましい。

■自社開発の内部エンジニア

- 足りない技術は外部に頼ってよい。ただし、最終的には自らができるように技術を使いこなせるようになれ
- 既存の IT とは別に、内製に強いリソースを別会社化。業界を変えるデジタルカンパニーとして位置づけることで、優秀な人材を集めよ

デジタルカンパニーを目指すにあたって、専門性の高い内部エンジニアを整備することは有効な取り組みではある。しかしながら、優秀なデジタル人材の獲得は容易なことではない。故に、今回の調査対象である先進企業も様々な取り組みを行っていた。例えば、育成という観点では、外部企業と手を組み現場をサポートしてもらい、足りない技術・ケイ

パビリティを補完しつつ徐々にスキル移転することを行う、といった方法も内部エンジニアの整備には有効な取り組みだ。また、人材獲得においては、理念や技術先進性に共感する優秀な人材を集めるために、デジタルに特化した別会社を立ち上げるといった方法も、有効である。

■外部リソースの活用

- スタートアップやベンチャーとの接点を積極的に作り、それらのケイパビリティを見極めろ
- 自社だけでできることは限られている。変革のビジョンを共有する会社とエコシステムを形成せよ
- パートナリシップ選定のポイントは、結局は人。一緒に仕事がしたいと思う信頼できる人がいる会社のその人自身が参画してくれるかどうかを確認しろ
- より大きな変革を実現するためには、ロビー活動も怠るな。国を巻き込んで制度から変革しろ
 - 海外企業の取り組みであるが、市場の変革や社会の変革といったより大きな変革を実現するために、ロビー活動を行うことで、国を巻き込んで制度から変革することに取り組んでいる企業もあった。

自社開発の内部エンジニアの整備がされていなければ、外部リソースを活用することでトランスフォーメーションを進めることもできる。ただし、双方がメリットを享受できる形で、パートナーシップを組むことが重要だ。今回の調査企業の共通認識として、「自社だけでできることは限られており、足りないスキルやケイパビリティの補完は外部活用を視野に入れるべき」という考えが見られた。実際に、変革のビジョンを共有できる会社とエコシステムを形成することができれば、スピード感を持って事業を進めることが可能だ。両者明確な目的意識さえあれば、スタートアップやベンチャー企業に限らず、海外企業まで、企業の状況に接点を積極的に作ることも有効である。

■開発・運用の内製化

- 内製化とは自社でプロダクトをコントロールすること。それが出来れば外部エンジニアを活用しても問題ない

“自社開発の内部エンジニア”で述べた通り、優秀なデジタル人材の獲得はすぐにできるものではない。あくまで、高いアジリティを実現するために必要な観点は、プロダクトの

コントロールを自社で行うことであり、足りないスキルを優秀な外部エンジニアに頼ることは問題ない。

【技術】

ITの世界において、技術の移り変わりは激しい。故に、”新技術の検討・取り込み”に関して苦勞している企業は少なくないだろう。同様に、”新しい開発手法の適用”に関しても、企業の取り組み方によって最適解が異なるため、悩みの種の一つだと推察する。共通する点は、いずれも世の中の最新とされている概念を取り込むのではなく、企業の事業や取り組み方に応じて、柔軟に選択する必要があるということだ。以下では、必要な技術や手法の選択に有用な、2つの指標についてのキーメッセージを紹介する。

■新しい開発手法の適用

- スピードを重視すべき機能と品質を重視すべき機能を切り分けて、それぞれに適した開発手法を使い分けろ
- 現場の要求変更をその場で対応して見せることで、スピード感を実感してもらえ
- DXプロジェクトでは、リリーススピードを重視した意思決定や優先順位のつけ方を徹底しろ

新しい開発手法と聞くと、アジャイル開発を思い浮かべる読者もいるだろう。実際に、アジャイル開発を用い、スピード感を持った取り組みを行うことで、現場からの理解や期待を高めることもある。また、開発手法に限らず、日常の意思決定や優先順位にアジャイルマインドを適用することで、高いアジリティの実現を目指す企業もあるだろう。

だが、アジャイル開発を導入している先進企業でも、開発手法においては、状況に応じてウォーターフォールとアジャイルを使い分けている。例えば、新規アプリケーションの構築に際し、初期段階ではウォーターフォール型のアプローチで行い、サービスを開始しユーザーエクスペリエンスを向上させる際は、アジャイル型へ移行し、開発手法を使い分ける、といったプロセス毎に手法を変えることが求められることもある。

■新技術の検討・取込み

- 全てのテクノロジーを自社内で構築することは不可能。それが本当に必要で重要なものを内製し、そうでなければ外部を活用しろ
- 丸投げではなく、圧倒的なスピード感でDXを進めるために、足りないケーバビリティを外部調達してコントロールせよ
- 自分たちのビジネスで競争力をつけるために、どんな技術が必要か目利きできる能力を身に着けろ

DX時代では市場やユーザー個人のニーズの動きに機敏に反応する必要が出てきた。いわゆるアジリティやスピードである。日々の中でいかに素早く反応できるかということや、今日時点での最も重要な課題や優先すべきアクションが何であるかを常に考え把握していなければならなくなったという事である。とはいえ、本来の業務と並行し、技術に対してあらゆることを内製で行うのは困難である。外部の知見やノウハウ、技術力を活用することや、協業しつつも吸収しスキルのトランスファーを常に図ることが競争力を強化することにつながる。

2.3. DX を継続的に進めるための考え方のまとめ

ここまでは、「変革規模」と「組織成熟度」という2つの軸を定義し、DXを進める上で有効なメッセージや施策を整理して紹介してきた。また、調査の中で、DXを継続的に実施している企業は、その時々の変革規模に応じた組織成熟度指標に関する取り組みをしていることがわかった。この変革規模に応じた注力すべき組織成熟度指標は、前節で紹介した三つのアクション(“①トップダウンとボトムアップの両面から火をつける”、“②活動を全社に波及させるための身体作りを行う”、“③デジタルカンパニーの流儀を体得する”)にまとめることができる。変革規模と組織成熟度、そして三つのアクションの関係を整理した図表が、図 2.3.1 である。

“①のアクションには、DXを始めたい企業、とりわけオペティマイゼーションをこれらから取り組もうとしている企業が注力していた組織成熟度指標がまとまっている。例えば、ビジョンの共有や、アイデア創出の仕掛けなどの指標が含まれているが、DXを始めようとしては避けては通れない指標である。これらの指標を上げる取り組みを行うことで、DXを推進させ、変革規模を上げるための土台が出来上がるだろう。

DXを行うための土台ができあがった後は、①のアクションに加えて、②のアクションによって取り組みを全社的に進めていくのがよいだろう。②のアクションにまとまっている指標群を上げる取り組みは、オペティマイゼーションを更に進める際にも、トランスフォーメーションを行っていく場合にも有効になってくる。

変革規模をより高めるには、①②のアクションに加えて、③のアクションに相当するより高度な取り組みが求められる。技術を活用し、DXの取り組みを自社が目指すビジョンに合わせて最適化することで、“(5)市場での立ち位置の変革や”、“(6)社会の変革”を目指すことができるだろう。

このように、現状の変革規模を踏まえながら、各企業の目指すビジョンに合わせて取り組み方を設定する必要がある。その際に、本書で示した目指すべき変革規模に応じたアクションを、是非とっていただきたい。

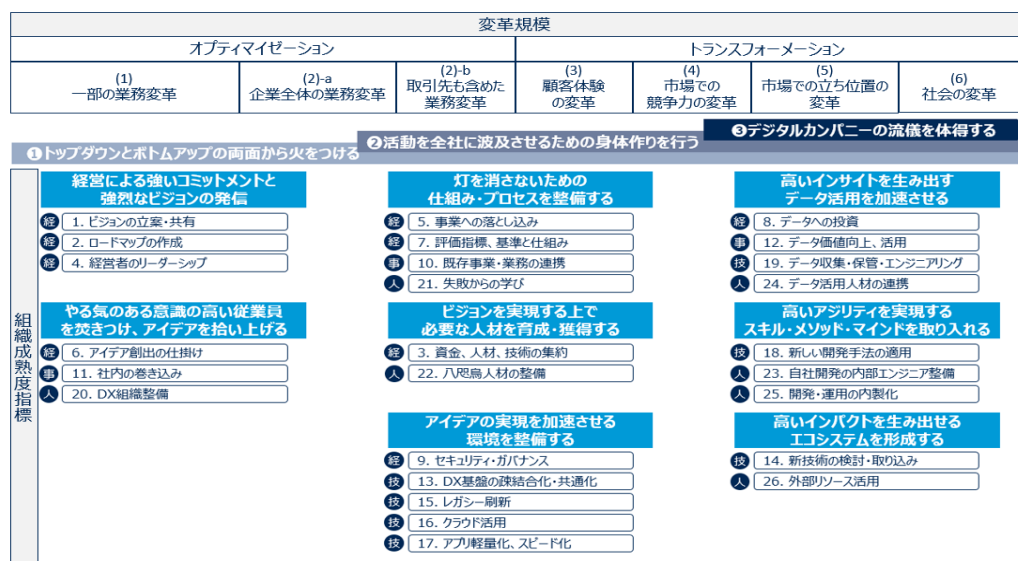


図 2.3.1 変革規模と組織成熟度の関係

また、「DX の継続的な取り組み事例に関する調査 概要報告書」では、本章の内容に加え、前述した三つのアクション毎に、企業が持つべき能力であるケイパビリティを整理している。例えば、ビジョンの立案・共有という組織成熟度指標を例に出すと、①の段階ではビジョンの視覚化する能力が必要だが、③を目指すには、世の中のトレンドも踏まえつつ、さらなる変革イメージをビジョンに反映する能力が求められる。

こちらを参照し、ケイパビリティの中から企業に不足しているものは何かを見定め、ケイパビリティの獲得を目指すことで、徐々に組織成熟度が向上し、各企業が目指す DX の実現の一助となれば幸いである。

2.4. DX の実践における課題の対応事例

ここまでは、DX を継続的に進めるための具体的な対応とそれを支える考え方について、DX 先進事例の広範な調査を基にまとめた。

しかしながら、現在 DX を先進的に進められている企業においても、DX にまつわる取り組みや施策が問題なく進んだわけではない。先んじて大きな手法的見通しがあり、それを遂行することで順調に DX 先進例に達した企業は一つとしてなく、漠然とした目標設定の中で試行錯誤を繰り返しながらはつきりとしたノウハウに達した企業、見通しは正しくとも実践段階で多くの試行錯誤を乗り越えてようやく DX を実現できた企業がほとんどである。本手引書の読者も、提示した考え方を実践していく中で、多くの課題や問題に直面することと考える。

したがって、過去の DX 先進事例において、DX の実現を目指す中で、どのような点が課題としてあり、どのような取り組みを行うことで、その後の DX の実現に結びつけることができたの

かといった情報は、有益なものであらうと考える。

実際に、課題にぶつかった様々な企業の取り組みを見てみると、課題を乗り越える際にも前節までで述べたような組織成熟度指標の向上が有用であった。

そこで本節では、DX の施策を行うにあたって、課題を乗り越えた経験をした 5 つの先行事例を紹介する。DX 推進における課題及び、課題を乗り越えるための取り組みと結果を整理した。具体的には、「業務変革を可能にするシステム開発を実現するうえでの課題を乗り越えた事例」を 3 社、「社内でデジタル化を活用した変革を行うための地盤を固めるための課題を乗り越えた事例」を 2 社で構成している。それぞれの事例に対し、DX 推進における課題及び、課題を乗り越えるための取り組みと結果を整理し説明することにする。各事例の課題は、どの企業もぶつかりうる内容かつ、乗り越えるための観点も様々な場面で活かせるものとなっているので、以下の試行錯誤の事例を示して本章の締めくくりとしたい。

(1) 業務の電子化が進まなかった(製造業 A 社)

<事例概要>

[課題]

従業員数 100 人未満のプレス加工を中心とした製造業の A 社は、作業の内容記録を手書きで倉庫に保管していた。製造業は製造物責任法によって製造の記録や品質管理データを記録し、これらの作業内容記録の保管が義務付けられている。保管期間は企業によって異なり、A 社は 10 年分の製造記録の保管を行っている。この作業記録管理業務は、作業毎に手書きで行われており、また紙媒体での保管なので、記録面でも参照面でも非効率的であった。

そんな中、A 社は様々な作業効率化ツールの普及と共に、作業内容記録の電子化や、効率化のためのシステム導入を模索していたが、製造業である自社の作業記録管理業務にマッチするソリューションが市販ではほとんど存在していなかった。また、作業内容記録の電子化のためにノート PC を導入した過去があるが、現場に定着しなかった。現場に定着しなかった背景として、ノート PC の操作が、作業の途中に入ること、作業の流れが中断されてしまい、作業効率を最優先とする製造業の業務に即さず、従業員にとって非常にストレスフルなものだったということがある。これらの点により A 社では中々 IT 活用が進まず、従来の紙媒体での内容記録作業及び保管を行っていた。

[取り組み]

A 社の経営者は自社に合った形で作業記録管理業務を IT 化できる開発パートナーを積極的に探していたが、コンサルティングを必須事項とするなど、A 社から見ると余分な内容を含む、高コストな提案が大半であった。

都道府県が主催する DX 支援事業で提案されたプランの中に、アジャイル開発の手法を用いて、段階的に IT システムを構築していく案があった。これは過去に依頼したベンダーの提案に対して、短期間かつ低コストで MVP (Minimum Viable Product : ユーザーに価値を提供できる最小限のプロダクト) を作成し、要件を反映させながらアジャイルで開発を進めるという、今までの他社の提案と異なるものであった。A 社は、この方法であれば、自社に合った形で、作業記録

管理の効率化をするための IT システムを作ることができるのではないかと判断し、これを提案した事業者を開発パートナーとして、作業記録管理システムの開発を委託した。A 社からは、現場の負担を考慮し、当初は経営者が一人で開発プロジェクトに参加した。

実際に MVP を作成した後は、過去のノート PC の導入失敗を踏まえ、まずは業務に精通したリーダー陣にシステムを展開し、実際に業務で使うことにより得たフィードバックを受け改良を加えることで、現場に受け入れられる試作版の提供を可能にした。具体的には、スマートフォンと QR コードを利用して 1 アクションで作業記録を可能にする工夫や、複雑な機能の実装を避けること、現場の生産性を優先することなどを最重要指標として評価し、開発する機能の厳選を行った。

経営者の主導 により行われたこれらの取り組みの流れを組織成熟度で表すと、下記の図のようになる。A 社としてはデジタルカンパニーの土台はないが、” 外部リソースの活用” と、それによる” 新しい開発手法の適用” によって、DX の段階を進めることができた。

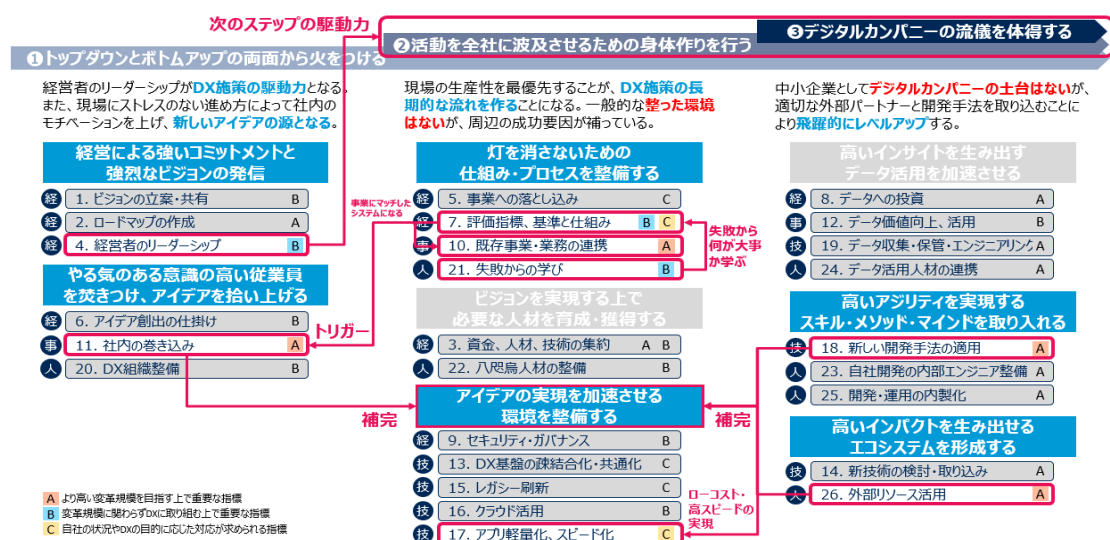


図 2.4.1 製造業 A 社の取り組み

[結果]

現場を巻き込み、従業員の声を反映させながら必要な分だけ機能拡張を繰り返すことで、今まで提案されてきた IT システムと比べ、低コスト・高スピードかつ現場に即した形でシステム導入を完遂した。

また、現場のフィードバックを短期間で反映することで、従業員がペーパーレスによる業務の効率化を通して、IT を用いた業務変革の利点を実感することに繋がり、現場の理解を得た状態でのシステム導入を可能にした。このことにより、現場から作業記録管理システムに対して、リクエストや機能要求も挙がるようになり、従業員がリクエストした機能を実装することによって、作業記録管理が簡単にできるようになるだけでなく、現場の生産性そのものが向上する効果もあった。

なお、A 社はその後、機能拡張を繰り返しシステムの汎用化が進んだことで、結果として本シ

システムを自社で利用するだけにとどまらず、A 社の商品として外販することも成功させている。

[本事例の要点]

- ・アジャイル&スモールスタートで、無理な長期的ロードマップはなくとも、事業にマッチしたシステムを作り上げられる。
- ・特に中小製造業等で DX を現場に受け入れてもらえるようにするには、システム等の使い易さやストレスフリーが最重要。
- ・予算等のリソースが限られている場合は、優先すべき選定基準を明確にしたうえでマッチしたパートナーを選ぶのが大事。

本事例の課題に対する取り組みと、その結果をまとめると下記の図のようになる。

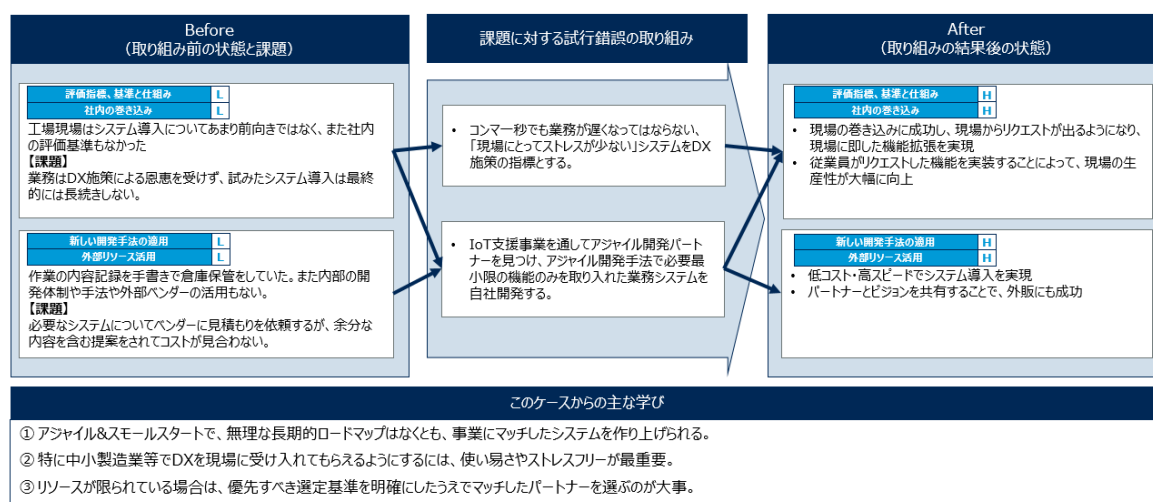


図 2.4.2 製造業 A 社のまとめ

(2-a)システム開発に技術的な壁が存在した(製造業 B 社)

<事例概要>

[課題]

溶接技術を専門とする製造業の B 社は、板金加工業において依然として職人の技術に頼らざるを得ない作業がある中、職人の高齢化に伴う人材不足といった経営課題を抱えていた。また、溶接技能に関しての技術の習得は、溶接動作の身体的な動作や、作業中の金属の望ましい状態を言語化することが難しいため、人から人への伝承によってしか行うことができていなかった。そこで、溶接技術における人材の確保と育成が急務である危機感から、熟練者と初級者の動作を座標によってデータ化して比較し、技能評価をレポートする機能や金属の状態の可視化機能を持った、溶接訓練支援システムの開発の検討を始めた。

溶接訓練支援システムの必要な機能について漠然とした構想はあったが、それらを実現させるための資金的、技術的なハードルは高かった。溶接作業の記録には、高価な機材が必要になってくる。更に、複雑な機材を活用したうえで、言語化の難しいベテランの経験や感覚をデータ化し、意味のある形で分析、蓄積をしなければいけない。このような壁がある中、B 社はどの様に溶接

訓練支援システムを実現するか模索していた。

[取り組み]

B社は本システムの5年以上前から、同業3社によるITに関する勉強会を開くなど、地道な人材育成を行っていた。時に専門家の手を借りながら、簡易的なプログラムなどITに対する素養を持つデジタル人材を育成することで、自社開発でローコードを用いた生産管理システムの実現や、受注や調達など一連の業務のシステム化を成功させている。本システムの構築においても、勉強会を行っていた3社で協力体制を組み、ソフト面やハード面といった各社の強みを活かした形で溶接訓練支援システム構築を目指した。B社は主にソフト面を担当し、実際の開発は経営者と、前述の地道な育成によってITスキルを身に着けた社内のデジタル人材によって行われた。

協力体制をとった3社は、限られた資金の中でシステムの構築に向けて、小型シングルボードコンピュータモジュールや、ビデオカメラといった低コストな機材を用いてPoCを繰り返していた。技術的な課題を見つけては、できる範囲で解決するようなトライアンドエラーを2年間繰り返すことで、システムを実現させるために必要な機材が分かり、より具体化させることができるようになっていた。

この様な取り組みの中で、高価な機材を用いなければ乗り越えられない課題が生じたが、自治体の公募型の研究事業によって支援金を得られる制度を知って応募することとした。支援金を得るための応募には溶接訓練支援システムを実現するための具体的な計画が必要だったが、2年間の試行錯誤によって構想が現実的な計画になっていたことが公的援助を得ることを可能にし、資金不足という課題を解決することができた。

資金を得て高価な機材を利用可能になった後も、B社及び協力会社の試行錯誤は続いた。技術面で、様々な小さな課題が生じたが、3社で週に複数の会議体を持ち、各社の得意分野を活かしながら効率よく課題を乗り越えていった。例えば、溶接時に強い光が出るためモーションキャプチャで動作のデータが取れない、作業時に生じる高圧電流によってシステム機器がショートする等といった技術的な課題も、3社での意見交換と、公募によって得た資金による機材の導入など、地道なトライアンドエラーを繰り返して乗り越えた。また、作業者の動きをデータとして取り込んだ後のデータ解析の経験等は無かったが、公的な産学官ネットワークを活用し、各分野の専門家からのアドバイスを求めることで、問題に対処し、取り組みを進めることができた。

この様にして、ハード面とソフト面の課題を着実に乗り越えることで、定着可能な水準まで、溶接訓練支援システムの精度を上げることに成功した。

これらの取り組みを、2.3で提示したアクションや組織成熟度を基に表すと、下記の図のようになる。

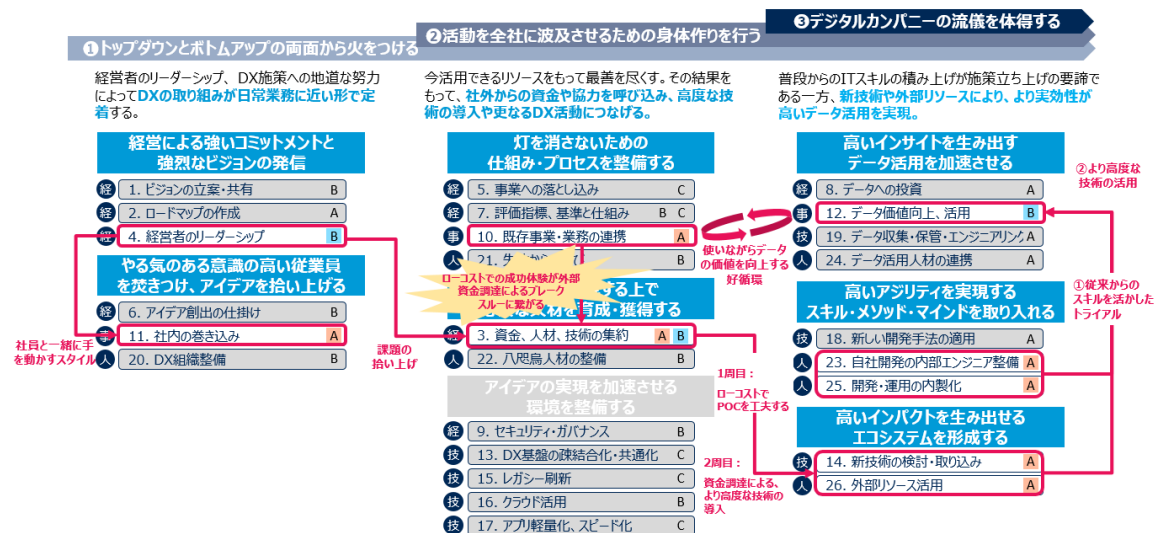


図 2.4.3 製造業 B 社の取り組み

[結果]

開発した溶接技術の訓練支援システムにより、訓練生の溶接技術に対する熟練に要する期間が従来の半分になった、と現場が認識する程の成果が生まれた。更に、言語化ができていなかった技能レベルの評価が可能になり、訓練する中での課題が明確化され、訓練生の技能向上のモチベーションも上がることになった。

公的支援の期間は終了したが、開発メンバーによって改善が続けられており、新たな生産管理の仕組みを作る計画を立てるなど、その後も活発に業務変革が行われている。

[本事例の要点]

- ・資金や人材に限られた場合でも、今あるリソースを最大限に活用して PoC を実行していただくことがポイント。この取り組みから新しい技術やアプローチへのアイデアが生まれ、更なる発展に繋がる可能性が高まる。

- ・テクノロジーに関する課題を全て自社で解決することは不可能。自社が持つネットワークを活用し、課題解決のアイデアを見つけることができる。

- ・組織として業務理解に加えて、IT やデジタル関連の技術を理解し、知識や経験を積み上げることが重要。外部との勉強会などで積み上げた経験があったからこそ、外部のアドバイスを得ながら短期間でツール化し、データの活用を業務に定着できるレベルに至った。

本事例の課題に対する取り組みと、その結果をまとめると下記の図のようになる。

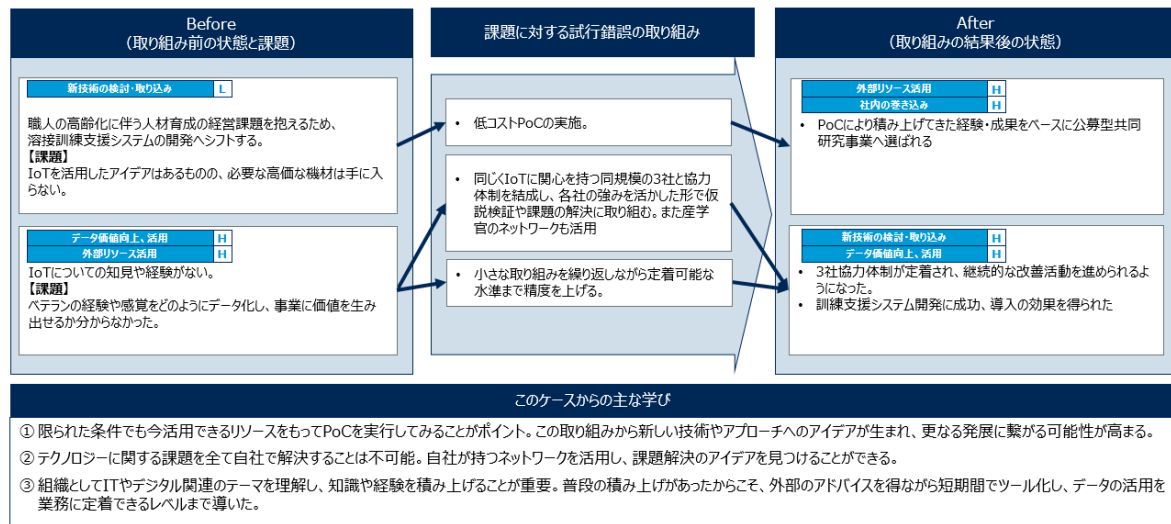


図 2.4.4 製造業 B 社のまとめ

(2-b)システム開発に技術的な壁が存在した(非鉄金属業 C 社)

<事例概要>

[課題]

大手非鉄金属メーカーの C 社は、使用済みの PC やスマートフォン(以降 e-scrap)を金や銀などの有価金属に製錬するリサイクル事業を行っている。近年、スマートフォンや IoT デバイスの普及に伴い電子基板需要が増える一方で、金や銀などの有価金属の資源不足が世界的にうたわれている。よって、e-scrap のリサイクル事業の市場規模は拡大しており、世界的に e-scrap の処理量はこの 10 年程で約 3 倍に拡大している。C 社はスクラップ業者から、商社を間に挟む形で e-scrap を買い取り、製錬を行っていた。C 社とエンドユーザーの間に商社を挟む理由としては、使用している自社開発のリサイクル事業の取引システムが、機能面や画面構成などに、知識を要し直感的に操作方法が分かるような使い勝手のよい IT システムではなかったため、商社等の取引の中間に入る一部のユーザーしか扱えなかったためである。

e-scrap の需要が増える中で、各地のスクラップ業者等のエンドユーザーを、新規ユーザーとして獲得したいと考えた。しかし、新しく取り込みたいユーザー企業は年配の方も多く、今までの基幹系システムに近い使い勝手の取引システムでは使いこなせない。海外展開や対象ユーザーの拡大を見据えると UI/UX や機能面の観点で刷新が必要だったが、既存システムの改修ではそうした対応を実現することが難しく、また、新たな取引システムを構築するために必要となる UI/UX やアジャイル開発に関して、社内には経験者がいなかった。

[取り組み]

システム刷新により対象ユーザーを広げるため C 社は、UI/UX に強い開発パートナー 2 社と組み、アジャイル手法により次世代の取引システムの開発を行った。社内の役割分担としては、事業部がプロダクトオーナーとしてシステムの機能・ロードマップ・スケジュールに対して責任

を持ち、IT 部門はシステム完成の実現に向けて開発に取り組んだ。しかし、C 社側がアジャイル開発に理解不足な状態で始めたことや、創設 100 年以上経っている C 社とベンチャー企業であるパートナーの社内の決定速度の違いなどにより、アジャイル開発という手法をとることに関する問題が発生した。そのため、以下のような対応が必要になった。

①機能を段階的に組み込んでいく中で、一定期間毎に行ったユーザデモの際に、ウォーターフォール型開発のような完成した状態での機能の充足を求めてしまうことが起こり、ユーザートレーニングや勉強会を通して事業部と IT 部門/開発パートナー側でシステムに対する期待値についての認識を統一した。

②開発パートナーが SaaS 等の新技術の活用を提案したことに対し、C 社のセキュリティチェックのためスピーディに対応できないプロセスがあった。しかし、プロセスを変更し、最低限のチェックを行った上で、全てのチェックと並行で開発環境での導入を進めて、本格的な導入までに全チェックを終えるルールとし、スケジュールへの影響を与えないようにした。

③事業部のメンバーが「システム開発を依頼する側」と「依頼される側」という概念を持ち続けてしまっていたが、アジャイル開発では一体となって進める必要があるので、経営陣レベルまで課題をエスカレーションして共有し、事業側のリーダーを開発側にアサインするなどして、体制を変更し、そうした意識を徐々に改善した。

上記対応により、既存事業を巻き込んだ形で UI/UX やアジャイル開発といった不慣れな開発を、主体性を持ちながら外部リソースと連携を取りつつ行うことができた。本取り組みは、以下の図のように表すことができる。



図 2.4.5 非鉄金属業 C 社の取り組み

[結果]

新しいユーザーの使用にも耐えられる使いやすさを優先した UI/UX を実現することで、既存のユーザーであった商社のような中間会社だけでなく、リサイクル業者のようなエンドユーザーを獲得して事業拡大につなげることができた。また、C 社はアジャイル開発に対する理解が進み、IT 面での対応力が向上した。

[本事例の要点]

- ・アジャイル開発においては事業側の当事者意識を持った関与が必要不可欠。エンドユーザーの代役として必要な機能要求の洗い出し、検証、またロードマップの構想に対する主体性が求められる。
- ・新しい開発手法の採用には知見がある適切なパートナーを選ぶのが望ましい。また、手法にとらわれ過ぎず、適切に社風に応じてルールや開発手法をカスタマイズすることにより定着がスムーズになる。

本事例の課題に対する取り組みと、その結果をまとめると下記の図のようになる。

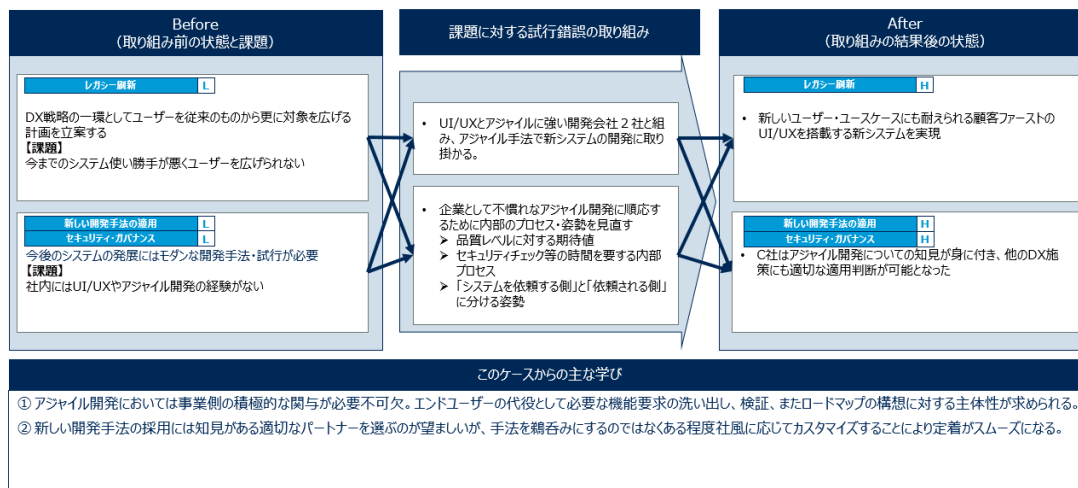


図 2.4.6 非鉄金属業 C 社のまとめ

(3)事業の変革を進めるための人材が不足している（サービス業 D 社）

<事例概要>

[課題]

サービス業 D 社は、教育や介護、キャリア支援等、異なる特性を持つ様々なサービス事業を多岐にわたって展開していた。D 社が幅広く展開している各事業の市場は、デジタル化の波を受けて同様の事業領域を持つ競争企業がクラウド化されたサービスの提供を開始するなど、競争が激化していた。一方 D 社は、デジタルを用いた事業の変革を推進していたが、現場こそが市場競争の将来を見通せるという考えから、エンジニアや PM といったデジタル人材の育成も含め、各事業部門に任せる形でこれを行っていた。しかし、実際には各事業部門自身が事業ニーズに合った採用や育成ができずにデジタル人材の不足をきたしており、ディスラプターへの対抗策も出せず、それどころか既存のプロジェクトも開発しきれず遅延を起こしている状況にあった。

この状況に危機感を持った経営陣は、外部の専門家を招き入れ、事業を横断するデジタル変革を推進する部署を立ち上げたが、この専門部署の横断的改善案は、事業部門毎に保有するデジタル能力や、事業の変革の進度によって必要な人材が異なっている等、実情にあわず、状況の打開はできていなかった。また、少数精鋭のデジタル人材の育成にも注力したが、各事業で必要なスキルセットが定義されておらず、D 社の企業規模が巨大なこともあって、全社的にインパクトを与えることができていない。

[取り組み]

D 社は、具体的な事業変革を各部署に任せることは引き続き堅持しつつ、これを担う人材を強化することと、その考え方を共通化させ、事業部を越えて刺激し合い、協力し合う体制を構築することで、これを活性化させることを企図した。

そこで、各事業に即した人材育成及び施策を行うために、事業を横断している全社的な IT 組

織に、事業部側のデジタル人材や、人事のリーダークラスを組織の中核メンバーとして迎えて、組織を再編した。

この新たな組織の下での共通フレームとして、各事業部門の事業変革の進度を明確化し、各事業部門が必要とするスキルや要員数を把握するために、D 社で実現すべき DX を以下の 3 つの段階で定義した。

①デジタルシフト(サービスや業務プロセスのデジタル化による、品質や生産性の向上を行う)(例：教育事業でのタブレットを用いた学習など)

②インテグレーション(オンライン、オフライン問わずに顧客本位のサービス提供を行う)(例:通学をオンラインに置き換えられるようにするための学習プラットフォームの構築など)

③ディスラプション(市場変化を見据えた新モデルを開発する)

このフレームの共通化によって、今まで一定の尺度が無く把握が困難であった事業毎の現状が可視化されると共に、事業の実情に合わせて今後のデジタル化に必要なデジタル人材の検討や、行うべき施策の検討も可能になった。

また、各事業のフェーズ（どのような DX に挑戦している段階か）に即した形でデジタル人材不足を解消するために、明確な定義がなかったデジタル人材のあり方について、自社に必要な役割を、全社的にエンジニア職やデータ職、企画職(PM 等)といった 7 職種に絞り込み、職種毎に 3 段階のスキルレベルを設定する形でフレームを設定した。更に、これらの職種とレベルを基に全従業員向けの IT 研修を実行し、従業員のスキルセットの可視化も実現した。これにより、各事業のフェーズを把握することで何をすべきかが明らかになり、そこで必要となるデジタル人材と社内にいる人材のスキルが可視化されたことによって、各事業部門はどのような人材がどのくらい不足しているかが具体的にわかるようになった。また、この具体的把握に基づいて、内部育成が可能な職種はリスクリングを行い、データ職など育成が難しい役割は外部に頼るなど、社内の現状に即した形で人材不足を補うことが可能になった。

IT 部門と事業各部門の状況に合わせ、DX を全社的に進められるようにするために、過去の失敗を乗り越え、事業の評価軸を作成し、IT 人材を育成、調達できる基盤を作り上げた D 社の取り組みの流れは、以下の図のように表せる。

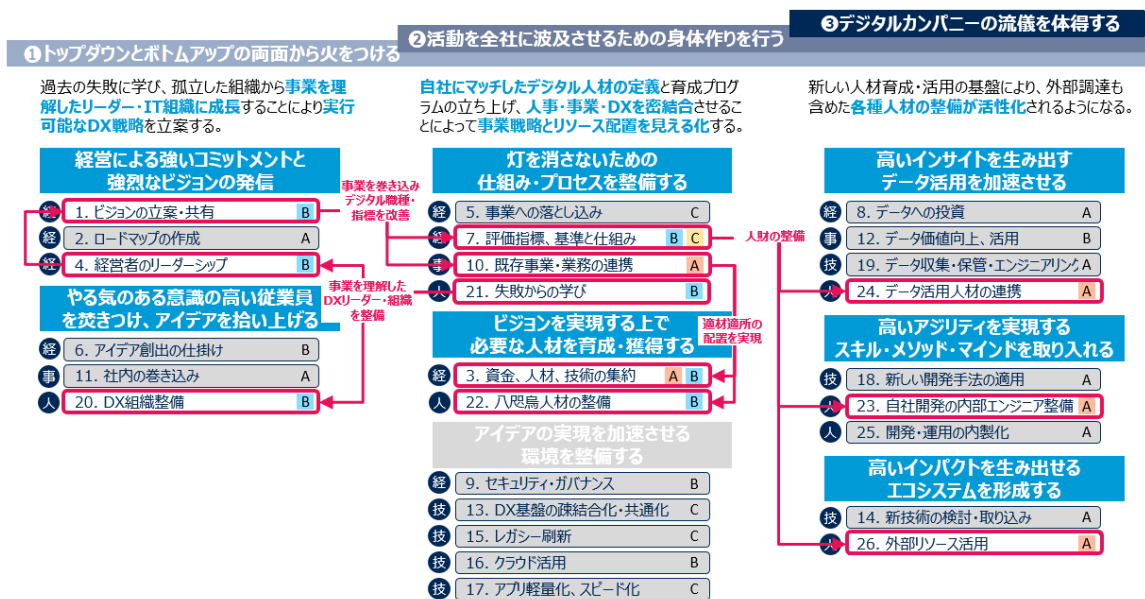


図 2.4.7 サービス業 D 社の取り組み

[結果]

事業のフェーズと、社内に必要なデジタル人材の可視化をすることで、リスクリングやアサインといった各事業部門の事業計画に合わせた形で人事計画が立てられるようになった。また、職種が簡素化されたことによって、社内で求められるデジタル人材が市場一般の言葉で表現することができ、社内での育成が困難な高度なデータサイエンティストや開発者等、足りていない人材の外部活用も、今までより容易になった。これらにより、事業にマッチした形でのデジタル人材の育成及び採用が可能になり、各事業を成長させる基盤が整った。

[本事例の要点]

- ・会社の体質・現状を理解したビジョンを立案しなければ DX 戦略は孤立し実現困難な「絵に描いた餅」になりかねない。DX 実現のリーダー・組織は事業を深く理解することにより実現可能なビジョンを立案できる。

- ・事業部門・人事部門等からの人材も DX 組織に取り入れることで、孤立した DX 組織から事業戦略の一部となった DX 組織に成長し、経営・事業・IT が相互で協力した形の DX 推進体制にシフトできる。

- ・人材の定義を行うことで、事業計画レベルで DX を進めるために必要となる人材のニーズを把握し、かつ全従業員の DX スキルレベルを可視化・数値化できれば、リスクリング・採用・アサイン等の人事計画が出来上がり適材適所の配置を実現できる。また事業側も事業計画達成の為、デジタル人材施策に積極的になる。

本事例の課題に対する取り組みと、その結果をまとめると下記の図のようになる。

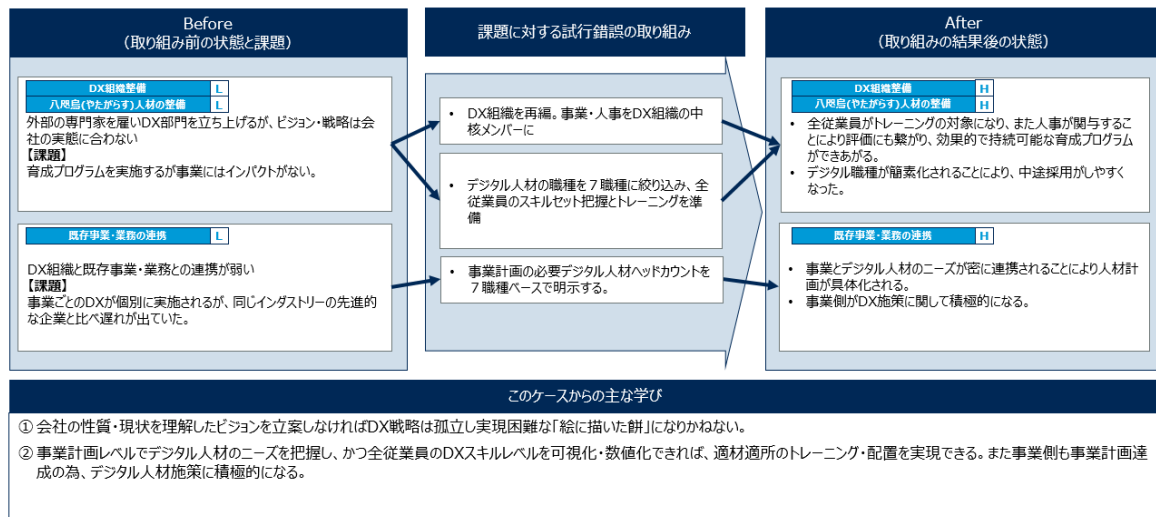


図 2.4.8 サービス業 D 社のまとめ

(4) トランスフォーメーションへの道筋が見通せなかった (化学工業 E 社)

<事例概要>

[課題]

大手化学メーカーの E 社は、自動車の軽量化部材などの機能商品、環境に対応した材料などの素材、医療用医薬品などのヘルスケアの 3 分野を中心に、グローバル市場で事業を展開してきた。いわゆるディスラプターの登場により、自動車業界など他の業界が脅威にさらされるという事態が起きる中、いずれ E 社が生業とする化学産業も同じ状況になるという危機感を抱いた経営陣は、グローバル市場の中で生き残っていくためにはデジタル技術を活用した業務の変革に取り組んでいく必要があると確信し、デジタル技術に通じた外部人材を雇用してこれを推進するためのデジタル部門を立ち上げた。

このデジタル部門は外部から来たメンバーを中心としていたため、まずは各事業部に受け入れられるように、現場での悩みや困りごとをヒアリングし、それをデジタル技術で解決できるようにサポートするというスタイルで業務を開始した。ここには、業務変革を推進するための組織はサポートと位置づけることで、業務の変革を起こすことそのものは E 社のそれぞれの事業の現場が主導するという役割分担を確立することも、もう一つのねらいとしてあった。そして、さらにそうした現場の動きが、新たな顧客価値、社会的価値を生む活動となることを、最終的な目的としていた。

ところが、実際には、各事業部門主導によるボトムアップの業務変革は足元の課題に対応するものに終始し、それで個々の部門の業務改善は実現できたものもあったが、あくまで個々の業務に視野が固定された状況では、全社的な新たな顧客価値、社会的価値を生む動きに繋がることはなかった。日常業務の改善と、事業の変革の間には大きなミッシングリンクがあり、このままでは最終的な目的には至らないのでは、という危機感が経営陣に生じていた。

[取り組み]

そうした課題に対応するため、異なる商材を扱う各事業部門が全社的な事業の変革につながる「発想」をもてるよう促すべく、デジタル部門は国内外のビジネス変革の事例をまとめた「アイデア集・事例集」を作成した。そのアイデア集・事例集において、事例は、ビジネスにデジタル技術を活用するための基本となる10個程度の考え方との組み合わせで説明されていた。具体的には、IoTなど既存製品にデジタル技術を掛け合わせて付加価値を高める事例や、ソーシャルメディア等様々なソースから得た知見を新製品開発やマーケティングに活かす事例などであるが、その基本となる考え方がいくつか組み合わせられることで新しいビジネスモデルを検討につながるよう、工夫して紹介されていた。

このアイデア集・事例集により、事業部門側は日常業務の枠から視点を広げることができ、事業変革のための考え方を、理解することができるようになった。また、各事業部門が扱う商品の特性なども異なっていたが、基本となる考え方を組み合わせることで、自分たちの事業の現状にあった、アイデアが生まれるようになった。

また、このアイデア集・事例集が、より実業務の中で使われ、効果を上げることができるようにするために、デジタル部門とそれぞれの事業部門で、ワークショップを開催した。そこでは、題材と一緒に取り組んで、業務変革に取り組むための考え方を体験できるようにした。その中で、このアイデア集・事例集は、コミュニケーションするための共通な土台として役割を果たし、参加者がそれぞれの見方を交換することでアイデア集・事例集の理解自体も進むこととなった。

そして、デジタル部門が、事業部門と現状課題の解決や現状業務に対する変革の検討の際にも、アイデア集・事例集をお互いの共通言語として利用することができ、検討案の実現に向けた、より深い議論を行うことができるようになった。

E社の取り組みを表すと、下記の図のようになる。

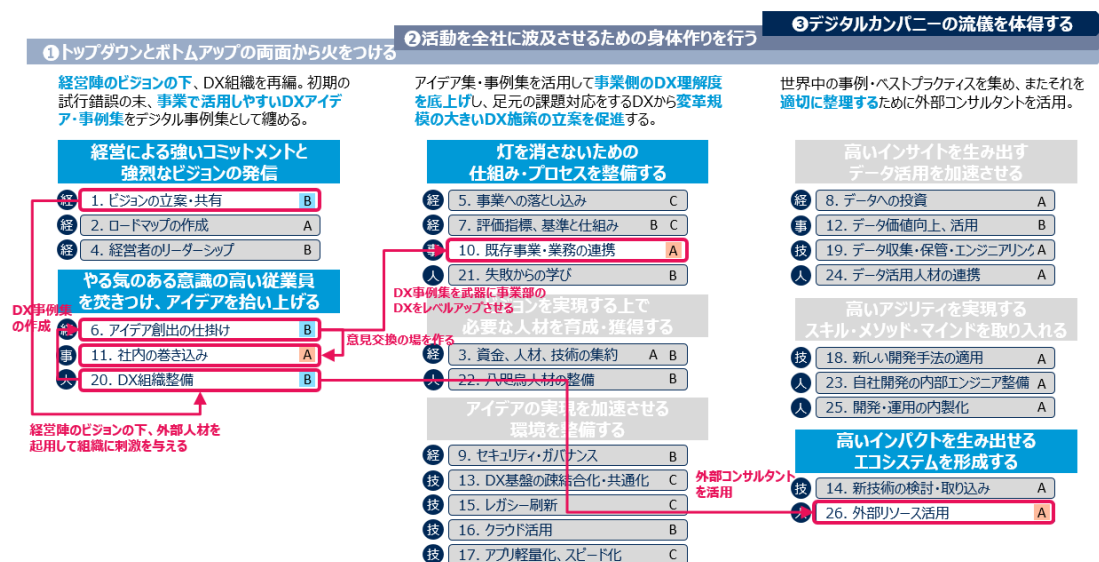


図 2.4.9 化学工業 E 社の取り組み

[結果]

アイデア・事例集を活用することで、一つの事業で行っていた取り組み規模が大きくなり、単に足元の課題を解決する改善だけではなく、複数の事業を跨ぐ業務変革につながるプロジェクトが提案されるような変化が生じた。これにより、例えば、複数事業のシステム統合によりデータを標準化することで、データ利活用をするための基盤を作るなど、より大規模かつ質の高い業務変革が実施されるようになった。アイデア・事例集という共通言語や、デジタル部門とのワークショップによる深い議論は、徐々にではあるが、確かに様々なプロジェクトで改善という枠を超えた取り組みに繋がっている。

この変化が新たな顧客価値や社会的価値を生み出す変革に結実したとはまだ言えない段階ではあるが、この取り組みを通して、事業の現場は最終的な消費者を意識する変革を起こせるような体質へと着実に変わってきている。

[本事例の要点]

- ・ボトムアップの DX 推進では事業の足元の課題に対する施策となることが多い。外部の情報や事例を事業部門に理解させていくことで、社内外に影響力のある変革規模の大きい取り組みも生まれやすくなる。
- ・外部人材によって成り立つような新設した DX 組織は、事業部門と密接な人的コミュニケーションをとらせる工夫が有用。これにより、孤立化を回避できるだけでなく、事業部門の信頼を獲得し、外部情報の理解を増進させることもできる。

本事例の課題に対する取り組みと、その結果をまとめると下記の図のようになる。

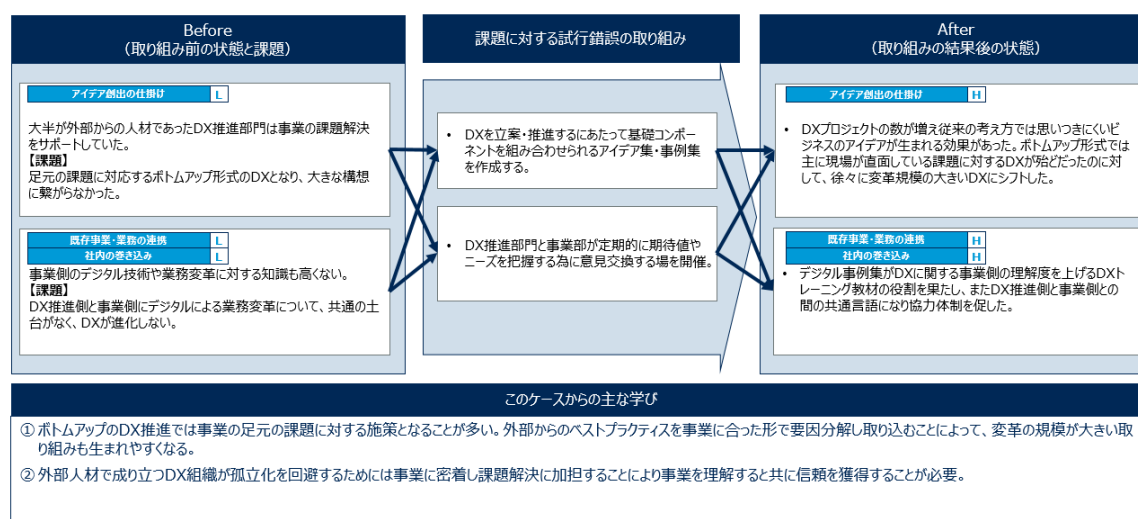


図 2.4.10 化学工業 E 社のまとめ

第3章 DX を実現するための IT システムのあるべき姿

前項までは、経営、事業、技術、人材・組織に関して、DX を実現し、継続的に進めるための考え方を述べてきた。

本項からは、その中で技術の観点について、DX を実現していくために必要と考える要件や個々の要素について、説明を行う。

まずは、先行事例企業への調査を通じて得られた、DX を実現するための IT システムの要素について共通するポイントを紹介する。

調査および整理の結果、先行事例において重点と考えられているポイントは、社内外の円滑かつ効率的なシステム間連携を目指す「社会最適」の観点、データ活用を中心に据えて社内外へ新たな価値を生み出してゆく「データ活用」の観点、IT システムとその開発運用の体制が変化に対して俊敏かつ柔軟に対応できる「スピード・アジリティ」の観定の3つに分類することができた。上記で「社会最適」という表現を使用したか、本書では競争領域、非競争領域を明確化し、非競争領域においてすでに社会に存在するリソース（外部サービス）を活用してビジネスを支えるシステムを構成し、そこで生まれる割り勘効果によって、自社の IT 投資額、開発・保守体制、リスク対策費用・人員を最適化することで、リソースを競争領域へ投下してビジネスを強化することを「社会最適」と記載している。

デジタルエンタープライズにおける IT システムの重要な視座がこの3つであることは、DX 推進指標の IT システム要件がまさにこれにあたることも傍証として観てよいかもしれない。

表 3.1 DX 推進指標「IT システムに求められる要素」

ITシステムに求められる要素 (DX推進指標 8)

データ活用 (DX推進指標 8.1) データを、リアルタイム等使いたい形で使えるITシステムとなっているか。	DXを推進する上では、そもそもどんなデータを持っているのか、その中でリアルタイムで使いたいデータは何かを認識し、実際にそれが使えているかが極めて重要であり、ITシステムはそれを実現できるものとなっていることが求められる。
スピード・アジリティ (DX推進指標 8.2) 環境変化に迅速に対応し、求められるデリバリースピードに対応できるITシステムとなっているか。	既存のITシステムの運用を前提としてしまうと、新規サービスのリリース頻度などを加速できなくなってしまう、結果として、加速する環境変化のスピードに取り残されてしまう。環境変化に迅速に対応できるようなデリバリーが可能になっているかがDX実現の上で重要である。
全社最適 (DX推進指標 8.3) 部門を超えてデータを活用し、バリューチェーンワイドで顧客視点での価値創出ができるよう、システム間を連携させるなどにより、全体最適を踏まえたITシステムとなっているか。	ディスラプターが顧客視点のビジネスモデルで攻めてきている中、それに対抗するためには、自社内外のバリューチェーンワイドでの組み替え等により価値を創出することが重要である。したがって、その基盤となるITシステムは、システム間連携などにより、部門を超えてのデータ活用が可能であり、また、データのオープンな流通を実現するエコシステムとの連携も容易に実現できるものとなっていることが必要となる。

DX 推進指標で述べられている「全社最適」は、部門を超えたデータ活用やシステム連携が前提として考えられているが、近年は「個社」という枠を超え、社会と繋がった IT システムや、

その連携を踏まえて、個社として最適であるための IT システム活用への発想が求められている。特に近年では、サプライチェーン全体を巻き込んだ共通の IT システム基盤や異業種間のデータ活用のための連携基盤も生み出されており、業界や国の枠を越えて、企業同士が共存してゆく仕組みが整えられてきている。その意味で、自社の IT システムアーキテクチャのグランドデザインを考察する上でも外部 IT リソースの活用は重要度が増しているものと考え、本章では「社会最適」という呼称、考え方に変更し、自社の IT システムの分類と、外部 IT システムとの協調・連携の視点を念頭に IT システムのあるべき全体像について検討する。

上記の IT システムに求められる 3 つの要件は、DX を推進するために全て欠かせないものであるが、それぞれの必要なレベル感は、各企業により様々である。すべての企業が各要件のレベルを最上級にしたハイエンドモデルを必要とするわけではなく、第 1 章で述べた通り、「ビジョンに即したもの」を目指せばよい。企業の IT システムそのものが足かせになることなく、変革を後押しする存在となるために、全社的な DX 推進への IT システム変革に向けたロードマップを策定する際のチェックポイントとして、本章を参照してほしい。

以下、この 3 つの要件について、詳細を説明していきたい。

3.1. 「社会最適」を実現するための IT システム要件

DX レポートでは、企業の IT 関連費用のうち 8 割以上が既存システムの運用・保守に充てられていると言及している¹⁵。DX の推進、すなわち、新しいデジタル技術を導入して、新たなビジネスモデルを創出するためには、IT 投資における「攻めの IT 投資」にシフトし、バリューアップに向けた取り組みを行うため、既存システムにかかる費用を軽減する必要がある。

特に近年は「デジタルディスラプター」と呼ばれる企業群が、顧客視点のビジネスモデルで、市場のニーズの変化に素早く対応し、さらには新たな市場の形成さえも行っている。こうした市場に対応するには、デジタルによる俊敏性・利便性を自社に取り入れ、自社が持つ強みをデジタルに対応した価値として創出することが求められている。そのためには、社内の IT システムアーキテクチャ、IT 投資状況とその将来像の全体を俯瞰して、自社が積極的にコストをかけて強化すべき競争領域と、外部の有用なリソースの活用も視野に入れ、不要なコストの削減を推し進めるべき非競争領域を的確に分化し、必要な部分は投資を続けるような、IT 投資計画を立てることが有効である。部門や個社を超えてデータ活用が可能であり、さらにデータのオープンな流通を実現するエコシステムとの連携、外部技術・外部リソースの活用を踏まえて、社内外を見渡し、社会最適なシステムを構築することが求められている。

- ① 非競争領域に外部の有用なサービスや、競争領域に最先端技術を取り入れる柔軟性があり、その分野で実績のあるベンダーやサービス提供事業者と価値を生み出すためのパートナーシップを提携できる

非競争領域に含まれる、(一般的に) 競争原理が直接働きにくい領域には、他社が開発したパッケージ製品や業界共通基盤などの導入が有効である。例えば人事・会計などのバックオフィスシステムなどは法改正や制度改正の外的要因、または組織改編や統廃合などの内的要因による要件変更が発生することは常であるが、これらの対応にもシステム追加コストを発生させないパッケージソフトウェアサービスが普及している。

日本企業の特性として「顧客第一の商習慣」に起因する、社内のエンドユーザーの意見が強いという、欧米諸国にないパッケージソフトウェア導入の難しさがある。エンドユーザーの意見を反映しようとする、基幹システムは、ベンダーの支援を受けて、スクラッチから書き起こす自社開発のもの、またはアドオン（機能拡張用のソフトウェアを結合させることで基幹システム本体の機能を増やすこと）が多いシステムにならざるをえない。それぞれの企業の事業形態や商習慣、場合によっては企業文化を反映した独自のシステムを構築するこ

¹⁵ 経済産業省 デジタルトランスフォーメーションに向けた研究会、「DX レポート～IT システム「2025 年の崖」の克服と DX の本格的な展開～」，2018 年 9 月，https://www.meti.go.jp/shingikai/mono_info_service/digital_transformation/pdf/20180907_03.pdf

となり、構築・運営・改修に膨大なコストがかかってしまう。企業や組織のビジネス活動などに関する全ての業務のパッケージが存在するわけではないが、多くのパッケージシステムが存在する汎用業務や経済活動の中で標準化された業務などでは、独自のシステムの開発や運用にかかわるコストを抑えることができる場合もある。

パッケージシステムを導入するためには、既存の業務をパッケージシステムの仕様やフローなどに合わせる必要も出てくるし、組織の構造や業務の特性によっては若干のカスタマイズが必要になるケースもある。エンドユーザーはこうした効率化や標準化のために、これまで長く慣れ親しんだ既存の業務のスタイルやフローを変えることに抵抗を示す場合もある。またパッケージシステムを導入したにもかかわらず独自性を優先し、結局のところ、業務の変化よりも、多くのカスタマイズによるパッケージシステムの改変で補おうとしてしまうケースも多い。最適化のために業務の変革を受け入れられなかったり、柔軟な対応ができなかったり、その結果として、システムのブラックボックス化やスパゲッティ化の温床となっている場合もある。

パッケージ製品の利活用にあたっては、エンドユーザーの業務を標準化し、製品が標準として提供する機能をできるだけそのまま利用することが望ましい。それによって、「既製品」を利用するメリットである、高品質・短期導入・低コストが実現できる。標準機能での運用により、パッケージベンダーがバージョンアップを行うと、自社のシステムも最新の状況に合った機能に更新される。このようなメリットを享受するために、自社固有の業務運用の実現要求とパッケージ製品を利用するメリットとを比較し、追加の作り込みやアドオン開発の規模を可能な限り抑えることが望ましい¹⁶。

なお、こうしたソフトウェアには SaaS の形態で提供される場合も増えている。SaaS の特徴的なメリットとして、費用ベースの請求となる「サブスクリプション」、「すぐに利用可能」、使用量に応じて金額を支払う「従量課金」という側面がある。SaaS のプレイヤーはビジネスのフロントからサポートを開始しているため、3つのメリットが、現場からのデジタル活用推進にも適合する。こうした特徴を持つ SaaS を組織として現場の主体的な導入・活用を推奨することも一考の余地がある。

競争領域への IT リソースを集中させるためにも、以上のような外部サービスの活用を視野に入れ、非競争領域のシステム開発・運用・維持/保守コストを削減し、競争領域における DX 推進や事業変革に取り組める体制を構築することは大いに検討すべきである。そのためには、まずは組織の IT アーキテクチャ・IT 投資状況全体を俯瞰し、自社の将来像・ビジョンと一致した IT システム戦略が必要である。

¹⁶ 独立行政法人情報処理推進機構、「システム再構築を成功に導くユーザガイド第2版」, 2018年2月, <https://www.ipa.go.jp/files/000057294.pdf>

競争領域における IT システム戦略については、部門または会社組織を超えたデータ活用を進めていく必要がある。つまり部分最適や局所最適をするのではなく、価値の連鎖の全体を俯瞰し、それを最適化していくような思考が必要となるわけだ。その結果としてビジネスとしてバリューチェーンワイドで顧客視点での価値創造ができるよう、サービスの拡充とリリースの俊敏性を高めることが求められる傾向にある。そのためには、外部のアプリケーションサービスやデータと連携する必要性が益々高まる。DX 推進指標では、部門間の枠組みを超えた「全社的なシステム連携」とされていたが、ここ数年での先行事例では、さらに会社組織の枠を超えて、データ活用に向けた取り組みや、サービスの連携を試みる潮流が見られる。IPA の上場企業を対象としたアンケート調査¹⁷によると、社外との連携の必要性について、半数以上の企業が「必要不可欠」と答え、8 割を超える企業が「必要」と回答している。また、社外との連携の目的として、約 8 割の企業が「自社にない技術力を獲得するため」という理由を回答している。

データの分析基盤、連携基盤、活用基盤、それぞれの分野で実績を持つソリューションベンダーやサービス提供事業者（プラットフォーマー）が国内にも着実に現れてきている。DX に関する技術の進化は早く、広範囲にわたっているため、領域ごとに得意とする事業者が異なる傾向にある。自社の DX テーマにおいて技術力や企画力、それを証明する実績のある企業を特定し、パートナーとして連携するためにも、外部環境の情報収集は怠らずに継続すべきである。

競争領域、非競争領域のいずれにおいても、IT システムのアーキテクチャとして、外部との接続仕様は、セキュリティを担保しつつ、世の中の標準的なものに合わせて、容易に連携できるようにしておくべきである。特に競争領域については、自社にノウハウの蓄積がない最先端の技術を柔軟に取り入れることを目的に、実績のあるベンダーとパートナーシップを提携することも考慮すべきである。将来に向けて、市場や技術の変化に応じて俊敏に連携・対応できることがあらかじめ考えられた構造になっていることが重要である。

¹⁷ 独立行政法人情報処理推進機構、「デジタル・トランスフォーメーション推進人材の機能と役割のあり方に関する調査」、2019 年 7 月、<https://www.ipa.go.jp/files/000073700.pdf>

3.2. 「データ活用」を実現するための IT システム要件

DX を推進するにあたり、事業環境の変化に俊敏に適応するためには、ビジネス上のニーズに合致するデータ活用と分析がより一層重要になる。

その背景には、多種多様なデータを集められるようになったことや、大規模データの処理速度が向上したことによる従来人間の手では難しかった膨大なデータの分析が可能になったことで、データ活用の選択肢が広がったことがある。多様なデータを組み合わせた分析、また高精度な予測によってビジネスを発達させてきた事例が世界中で増えてきたこともあるだろう。

収集したデータに基づいた分析・予測の結果を踏まえてビジネスの意思決定や課題の解決を迅速に行っていくことをデータドリブン（Data Driven）と言う。これまでの属人的な経験や勘に頼る業務プロセスを可能な限り廃して、大量のデータをリアルタイムに処理・分析し、昨今の複雑化した顧客行動の把握や現場業務の最適化、課題の発見などに活用することを標榜し、その重要性が叫ばれるものである。しかし、データドリブンな業務プロセスを取り入れるには、活用できるデータが揃っていないかったり、活用できる状態にデータが整備されていないかったり、データがあってもうまく連携ができないなど、旧来のシステムのままでは実現が難しい場合があり、組織全体の IT システムとしての見直しが必要である。

これまで多くの IT システムは、個別の業務や個別の用途を実現するために開発されてきた。これらのシステムで扱っている情報をもともの用途以外で用いようとすると、アクセスによる更新待ちが多く発生することや、目的の情報を含む大きなデータ群のまとまりでしかアクセスができないということが生じることがある。DX で求められる「データ活用」はこうした制限によって阻害されてしまう。

こうした制限を克服し、必要なデータを利用し、サービス活用や分析を高度化するプラットフォームとして注目されているのが、データ活用基盤である。データ活用基盤は、企業の内外のシステムから様々なデータを収集・蓄積・加工する機能を有し、利用しやすい形でデータを蓄積・保存する機能を有するものである。企業の基幹となる業務を管理する基幹システムや、DX の取り組みとして新規に構築したサービスそのものの負荷にならないよう、データ活用基盤を介して疎結合に連携させることで、それぞれが個別に開発・運用できる環境を用意することも、その役割である。

以下の図 3.2.1 は、データソースからデータ活用までのプロセスを、データ活用基盤を中心として図式化したものである。

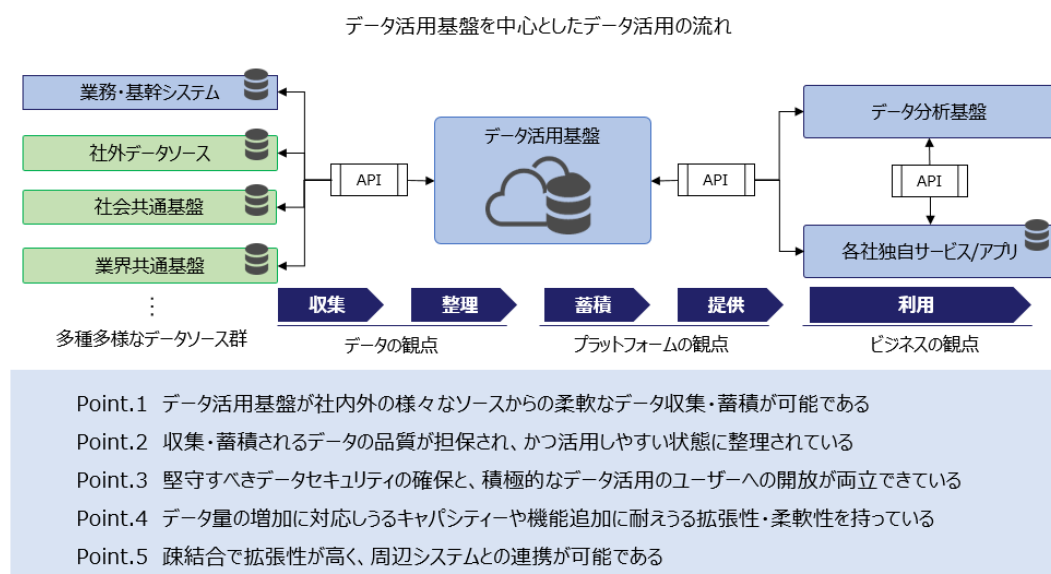


図 3.2.1 データ活用基盤を中心としたデータ活用の流れ

業務・基幹システム、および社外のデータソースなど、社内外の様々なデータを収集・整理・蓄積（保存）し、周辺システムに提供するプロセスにおいて、データの分析や新しいサービスへのデータの活用に柔軟にかつ俊敏に対応できることが重要である。そのために、予めデータ活用基盤を中心に据え、API などの疎結合な連携を実現する手法を採用することで、様々なデータを活用しやすいシステムを構築すると良い。そうして漸く、データドリブンな意思決定や業務プロセスが実現できる素地ができる。

データ活用において重要な点は、どのようなビジネス、用途に利用するのか、データを活用する目的を定めただうえで、バックキャストで上記のプロセスを検討していくことが必要である。データ活用基盤そのものの構築が目的化してしまい、組織全体のビジネスにおけるデータ活用の目的を見失うことは決してあってはならない。但し、データ活用のビジネス上の目的については、データ活用におけるシステムの要件というよりは、ビジネスの観点におけるデータ活用の要件であり、これについて語ることは本書の役割ではない。本節では、あくまで一般論として、データの観点とプラットフォームの観点を中心に図中に示した 5 つのポイントをもとに、データ活用における要件を示すことにとどめたい。

① データ活用基盤が社内外の様々なソースからの柔軟なデータ収集・蓄積が可能である

DX を実現するためのデータ活用においては、市場やニーズ、ビジネス環境の変化に応じて、必要とされるデータソースやデータの種類、その活用方法などが頻繁に変化する。社内だけでなく、社外データの活用ニーズも増え、多種多様なデータを組み合わせることで、データの価値は高まる。高度なデータ分析を志向する企業ほど、分析力だけでなく、データそのものを外部から調達する傾向が強くなってきている。そのため、必要に応じてデータを柔軟に収集する

ことを見越してデータ活用基盤を構築することが望ましい。

しかし、日本企業の現状として、IT システムは部門ごとに最適化され個別にシステム構築されてしまっているケースが多いため、部門やグループ企業間のデータ連携が難しい事情があるといわれる。加えて、企業の IT システム導入が進み、複雑化するにつれて、データは複数の IT システムに分散され、どこにどのようなデータがあるのかわからない、データ形式が異なっていて統合ができないなどといった新たな課題が生まれてくる。データは分析・活用して初めて価値を発揮するものである。社内、グループ企業内の各所に分散したデータを統合して整理・分析を行うためにも、データ活用基盤を中心に、データソースとなる社内外の各システムと API などを通じてデータ連携を行うことを念頭に置いた社内インフラの整備が求められる。

データを収集するにあたっては、まずデータを活用するビジネス上の目的が定まり、その上で必要なデータは何かを整理することが成功事例を生み出す近道である。ただし、データの収集目的や、収集するデータそのものは、市場の変化とともに、時系列で変化していくものであるため、データプラットフォームとして社内外のあらゆるデータソースと柔軟に連携していけるような仕組みが必要である。そのためにも、セキュリティを担保しつつ、世の中の標準的な形式や手順に対応できる、外部との接続機能を予め備えておくべきである。

② 収集・蓄積されるデータの品質が担保され、かつ活用しやすい状態に整理されている

データを活用してビジネスに利用するためには、データをただ集めればよいのではなく、集めたデータの品質を確保する必要がある。ソースから収集したデータは、当該データそのものだけでなく、統合・処理・活用といった流れを経るため、元のデータの品質が確保されていないと、どんなに途中の工程の改善を行ったとしても、データを利用したサービス全体の品質も低下してしまう。

データマネジメントの国際団体である DAMA (Data Management Association) が策定した DMBOK¹⁸によると、データの品質とは、正確性、完全性、一貫性、整合性、妥当性、適時性、一意性（重複排除）、有効性などによって評価されるとしている。

¹⁸ DAMA 日本支部, 「データマネジメント知識体系ガイド第二版 (DMBOK2)」, 日経 BP, 2018 年

表 3.2.1 データ品質の評価軸

品質の評価軸	説明
正確性	データが「現実」の実態を正しく表している程度を意味する。
完全性	必要なデータがすべて存在するかどうかを意味する。
一貫性	データ値がデータセット内やデータセット間で一貫して表現されている状態であり、データセット間で一貫して関連付けられている状態を意味する。
整合性	完全性、正確性、一貫性に関する考え方が含まれており、参照整合性（両方のオブジェクトに含まれる参照キーを介したデータオブジェクト間の一貫性）か、データセットが万全で欠落部分がないという内部的一貫性のいずれかを意味する。
妥当性	データパターンが期待を満たすかどうかを問うもの。例えば、地域内の顧客に関する既知の情報に基づいて、地域別の売り上げ分布が適切かどうかを判断する。妥当性に関する考え方には主観が含まれることがありデータ利用者と協力して、客観的な比較方法を策定するために、データに何を期待するか、それはなぜかを明確にする。
適時性	データ値が情報の最新版であるかどうかの評価。データが変化する頻度とその原因という観点で評価される。例えば、国コードのような静的な参照データの値は、長期間にわたって現在の状態を維持できるが、株価などのボラティリティの高いデータが現在の値を維持するのは短期間だけである。
一意性 (重複排除)	エンティティがデータセットの中に複数存在しないことを示す。
有効性	データ値が定義された値ドメインと一致するかどうかを意味する。

この点を踏まえて、データの品質が低い状態とは、例えば、ある一定の数値のみが許容される項目に、外部データの結合時に、そのルールを逸脱した数値データが入力されてしまい、処理にエラーが生じてしまったり（一貫性の欠如）、データの発生・収集から提供・利用までのリードタイムが長く、発生時には正しくても、利用する際には不適切なデータになってしまったり（適時性の欠如）と、様々な要因がある。

このように、ビジネス上の目的に沿ったものではない品質の低いデータをそのまま放置してしまうと、アウトプットとなる分析結果やサービス品質にマイナスの影響を及ぼしてしまう。そのため、より品質の高いデータを入手できるデータソースとデータプラットフォームの双方の観点からデータ品質の改善が必要となる。

ただし、品質を高めようとすればするほど、コストがかかることにも留意が必要である。データ自体の品質を保つためには、データ活用基盤のシステムの改善のみならず、機械処理もしくは人力での変換、補完、突合や、破損データや異常値データの除去などのデータクレンジング作業が必要になる。さらに、データ活用強化のためのデータ運用を検討する組織を組成し、データの品質管理と運用を専門とする組織が先行事例にも見られた。ただ闇雲にデータ品質を高めようするのではなく、データの発生・提供側と利用・運用側での目的を合わせた上で、それぞれの目的に応じたデータ品質管理を考えることが必要になる。

その留意点を踏まえた上で、全社的な DX を推進するための重要な要素として、現場でのデータ活用の促進のための3つの条件を紹介する。

1つ目は、頻繁に社内ユーザーが利用するデータに関しては、事前にプレ処理やクレンジング処理がおこなわれており、利用しやすい形式になっていることである。2つ目は、データを利用するのに有利なシステム環境条件が整っていることである。3つ目はデータの所在や形式、

データの持つ意味、時期やバージョンなどが統合的に整理された状態にしておくことである。

データ活用促進の3つの条件

- ① **事前にプレ処理やクレンジング処理がおこなわれ、
利用しやすい形式になっていること**
- ② **データを利用するのに有利なシステム環境条件が整っていること**
- ③ **データの所在や形式、データの持つ意味、時期やバージョンなどが
統合的に整理された状態にしておくこと**

ユーザーはますます大量化し多様化するデータ群から、適宜必要なデータにアクセスし抽出したうえで的確な処理を施さなければならない。ところが、そのデータに行き当たることさえ難しくなっている現実がある。データを起点にビジネスを改善し、加速させるためには、データの所在地やその定義などを明確にし、透明性を確保する必要がある。そのための仕組みとして、データの資産台帳となる情報が必要となる。この情報を「データカタログ」「データディレクトリ」「データレジストリ」などと表現されるが、ここでは「データカタログ」と表現する。「データカタログ」の整備はデータへのスムーズなアクセスを実現するのには極めて有効な手段である。

データカタログとは、組織内で保有するデータの定義や形式、要素などの情報を整理、可視化するための機能で、データの辞書とも言われる。社内システムの構造化データに加え、画像やセンサー情報などの非構造化データや、社外から調達したデータも管理の対象に含まれている。データを起点にビジネスを改善することを考える場合には、どのようなデータがどこにあるのか、検索性を高め、正確に把握することができる状態を保つ必要がある。そのためには、データカタログを整備し、かつ常に最新の状態に更新する仕掛けや組織体制も必要になる。

③ 堅守すべきデータセキュリティの確保と、積極的なデータ活用のユーザーへの開放が両立できている

データが利用しやすい状態に整理されていて、誰でもアクセスできるようにすることで、データ活用による価値創出の促進を後押しする必要がある一方で、漏えいなどのセキュリティ対策を行う必要もある。企業のセキュリティ事故やサイバー攻撃の被害が深刻化している昨今、DX の取り組みにおける積極的データ活用と、コーポレートガバナンスとしてのセキュリティ対策は一見相反するように見える。

デジタルサービスの企画構想の当初から、完璧なガバナンスやセキュリティを実現しようとすると、初期投資の負担が重く、プロジェクトを小さく始めることは困難となってしまう。もちろん個人情報や機密情報などの最低限堅守しなければならないデータやその保護を行う領域を予め定義し、絶対に漏洩してはいけないデータの機密性は担保すべきであるし、セキュリティの要素は、事業として外部にリリースされる前には慎重に考慮されるべきである。しか

し、それは全てのデータに対して最高レベルのセキュリティを施し、データの囲い込みを行うことではない。データそれぞれの持つ意味や不正利用のリスクなどを評価したうえで必要なセキュリティを施し、それぞれに的確なアクセス権を付与する。予めデータの取り扱いに対する線引きを定めた上で、ある一定レベルでの使い勝手を優先するような動きも、先行事例においては確認できた。データセキュリティに限らず、事業推進に関するガバナンスは、構想策定、立上げ期は事業の種をつぶさないよう、十分な配慮が必要である。

その一方で、DXの取り組みにおけるデータ活用において、(セキュリティや権限の問題をクリアしたうえで) より多くの社員がデータを起点として、価値を創出できるようになるような状態も目指すべき組織の姿である。これまでのデジタル人材育成の遅れやデータによる価値創出が一部の業種業態に限られている状況を鑑みると、誰もがデータを起点に価値を創出するような働きをすることを、直ちに求めることは難しい。しかし、「データが21世紀の原油である」とも言われる昨今、組織を構成する人材の一人でも多くが、データを自由自在に活用できるような人材育成と体制を構築していく必要がある。有能なデータサイエンティストを雇い、データを使って何が出来るかを社員に見せても、社員が自らデータにアクセスし、データを有効に活用できるよう、データの民主化を実現させなければ意味がない。「データを使って何かしたい、価値を生み出したい」と考える組織文化を作るためにも、データそのものの整理と利用環境の改善、そしてそれを守るべきセキュリティを堅守するための工夫は必要不可欠である。

④ データ量の増加に対応しうるキャパシティーや機能追加に耐えうる拡張性・柔軟性を持っている

DXの取り組みを継続していくためには、データ活用の試行・分析・検証・実践のプロセスを繰り返し行っていくことになる。技術の進歩とともにデータの起源や流通量、活用方法は増加し続けており、令和2年情報通信白書によると、我が国におけるデータ流通量は、急激なデジタル化の進展とともに拡大しつつあり、今後もさらに伸びていくと予想されている¹⁹。各企業のDXの取り組み過程においても、データの取扱量は増加の一途を辿ることが想定される。これらの社会的背景を見越して、現時点ですぐに利用することが難しいデータであっても、その性質上経年の蓄積に高い価値がある場合などは、今すぐに蓄積を開始し将来利用できるように準備をしている事例もある。こうした事例では経年のデータを資産としてとらえている。予測の精度向上や変化の過程を正確に把握するなど、10年後に20年分のデータが将来必要になり、必要になってから集めるのでは遅いといったことを想定したうえで、今できるアクションを起こしているのである。

¹⁹ 総務省「令和2年情報通信白書 第1部 5Gが促すデジタル変革と新たな日常の構築 第1節 5Gが加速させるデータ流通 1. データ流通量の爆発的拡大」, 2020年8月, <https://www.soumu.go.jp/johotsusintokei/whitepaper/ja/r02/pdf/02honpen.pdf>

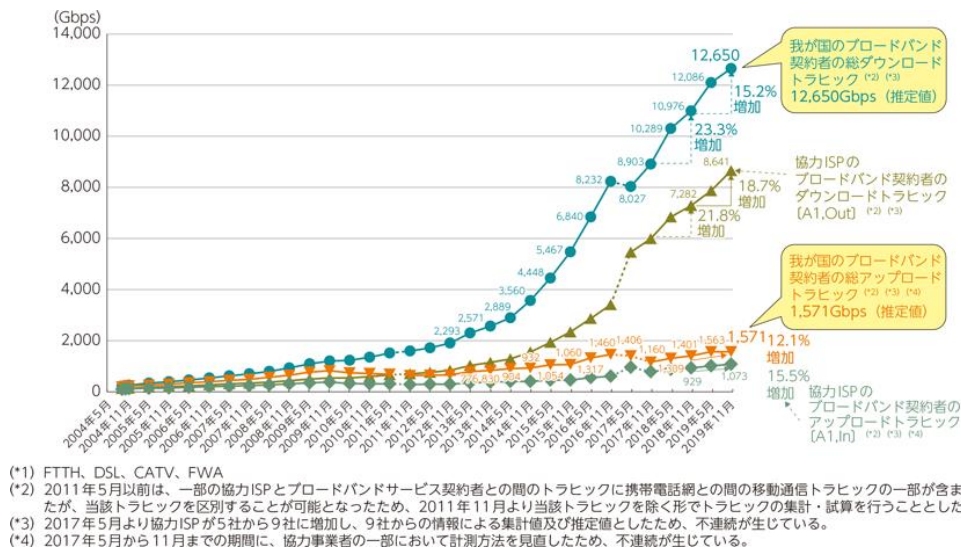


図 3.2.2 我が国のブロードバンド契約者の総トラフィック（令和2年情報通信白書より）

このような状況下では、今後各社で扱うデータは、接続するデータソースやデータ量が時間とともに増える中で、多数のシステムから継続的に収集され、増加していくことが予想される。増加するデータに合わせて、必要なリソースを必要なだけ自由に設置でき、パブリッククラウドのように利用した分だけ費用を支払う仕組みが注目を集めている。データ取扱量のキャパシティを拡張できる柔軟性を予め考慮したうえで、ビジネスの需要にあわせて、俊敏に増強できることが求められる。

⑤ 疎結合で拡張性が高く、周辺システムとの連携が可能である

変化の激しい市場において、データは必要な人に必要なタイミングで素早く渡していかなければ価値がなくなってしまう。データそのもののリアルタイム性と、仮説検証のための俊敏な分析・活用基盤の構築と接続が要求されている。データをより有用に活用し、高付加価値なサービスを生み出すために、どのデータをどのタイミングで誰に渡していくかの研究が各業界・各組織で今後活発化することが推察される。

そのためにも、データ活用基盤は周辺システムとの接続が容易に着脱できるよう予め備えられていることが、自社がデータ活用の流行に対応する上では望ましい。例えば、データ活用基盤と周辺システムを API でつなぐ仕組みを予め用意しておくなど、データのやりとりを密結合させず、疎結合にすることが有効な手段の一つである。

データのやり取りが疎結合な状態とは、細分化された個々のシステムの構成要素同士の結びつきが比較的緩やかで、独立性が高い状態のことである。システムの構成要素同士は連携しているが、双方の依存度は低い状態であるため、個々の機能追加や変更を加えやすく、柔軟な対応を行うことができる。

一方で、密結合している状態とは、プログラム A を改修すると、プログラム B を必然的に改修しなければならないような状態である。こうした密結合なシステムがもたらす結果として、仕様変更に伴う修正が徐々に属人化し、連携ロジックもブラックボックス化が進んでいくような状態が発生する。いわゆる「スパゲティ化」と言われる状況に陥った状態だが、このようなデータ同士が依存しあう密結合の状態は、新機能の追加や、変更などを難しくする傾向にあり、開発コストの増大化や、次項で述べる「スピード・アジリティ」への影響も大きい。

データ活用基盤全体として、増加するデータを適切に管理し、必要な業務の適正な情報のみが必要なタイミングで取り出せることが求められる。そのためには、データ活用基盤は構築してしまえば終わり、というわけではなく、市場の変化やニーズに応じて、データ活用基盤そのものの機能や状態の見直しを行う必要がある。この疎結合な仕組みを予め備えておくことで、新たなデータ活用の道筋が見えてきたとき、俊敏かつ最適な形での新機能の追加、また外部のシステムとの連携ができる。活用できそうなデータソースを適宜取り込み、ビジネスに応用するためにも、データ活用基盤そのものが疎結合で、柔軟性と拡張性を備えた設計が望ましい。

3.3. 「スピード・アジリティ」を実現するための IT システム要件

変化が激しく不確実な市場環境に俊敏に対応するために、求められるデリバリースピードに対応できる IT システムとなっているか。「スピード・アジリティ」はこの問いに答えるための要素である。

「スピード」と「アジリティ」の違いについては、様々な定義があるが、本書での定義は以下である。「スピード」は、一定の機能や品質を保ったシステムをどれだけ俊敏に構想・設計・開発・運用することができるか。「アジリティ」は、システムを構想・設計・開発・運用する際に、市場や環境の変化に応じて臨機応変に軌道修正できる柔軟性を示す。つまり、スタートからゴールまで一つの区切りの時間短縮に関する内容はスピードに該当し、スパンを短くする・こまめに軌道修正するなどといったサイクルを増やす、細かく分化したサイクルを繰り返すという臨機応変に対応する概念のものをアジリティに分類するのが適当である。

「スピード・アジリティ」とは、一言で言えば「変化対応力」そのものである。デジタルが浸透した現在、顧客や市場へのアウトプットとして、デジタルを活用した良いビジネスアイデアを思いついても、抱えている IT システムの柔軟性がなければ、その対応へのコストは嵩み、リリースまでのスピードも落ちてゆく。やっとの思いでサービスをリリースしたときには、既に他社に先を越されてしまい、後塵を拝するような事例も見受けられ、今後もデジタルビジネスの先進性や提供価値の格差は広がってゆくと推察される。システム単体が大きくなればなるほど、言い換えれば、一つのシステムが抱えている機能が多くなればなるほど、この要素を実現することは難しくなる傾向にあるが、以下に記載したそれぞれの要件を参照して、「変化対応力」のある自社のシステム構築に向けて役立ててほしい。

- ① アプリケーション（≒プログラム）同士が密結合せず、機能単位で疎結合に分離・独立しており、API 等の連携するための技術を活用して、接続/切断が容易に行えるようになっている

DX を実現するには、事業環境の変化に俊敏に適応し、ビジネスサイクルを素早く回す仕組みを整えることが求められている。社会や市場、顧客ニーズの変化に対応し、素早くかつ低コストでデジタル起点のサービスをデリバリーしていくためには、IT システム全体を見直す必要がある。

DX の実践に適しているシステムは、ビジネスモデルやサービスの個々の変化に応じ独立に俊敏かつ安全にアプリケーション/プログラムを更新できるシステムである。従来型のシステムでは、一部の機能改変や機能追加があると、想定外の処理が行われることの無いよう、全ての機能を網羅的にテストし直すなどの工程が必要となり、サービス開発・リリースまでに時間とコストがかかっていた。これでは DX に求められる俊敏なリリースが難しく、DX に適したシ

システムとは言えない。追加開発したシステムが既存システムに容易に接続でき、一部の機能を改変・追加しても、他のシステム・機能への影響が最小限になるような、各機能が疎結合に分離・独立しているシステムが良い。IT システムが持つ機能がサービスの単位で細分化され、独立にデプロイできることで、テスト範囲が限定されていて時間とコストが少なく済むからである。また、機能が疎に分割されていると、その機能ごとに十分なテストを行いやすく、かつ機能内部での更新があっても他の機能に影響を与えることはない。こうした前提に立つと、機能ごとのデプロイを行っても全体としても十分な品質が確保できると考えられる。DevOps のように、ビルドテストなどの開発の一部や運用の自動化を行うことも重要である。

機能単位で疎結合に分離・独立しているアプリケーション群をそれぞれ連携させるにあたっては、新規に構築した DX 基盤/アプリケーション、基幹システムそれぞれの負荷にならないよう、密結合させずに「緩く」連携させる工夫が必要である。API 等を活用することで、アプリケーション間、データを疎結合に連携できる。企業内で API 基盤を構築し、マイクロサービス間、及びモノリシックな既存システムとの疎結合な連携を実現することも、DX を先進的に推進する企業の事例として存在している。

先述したデータ活用でも述べているが、共通のデータ活用基盤を用意し、アプリケーション同士を密結合しないように切り分け、それぞれが独立した環境を用意することで、アプリケーションが必要に応じて様々なデータソースに容易にアクセスできる。必要に応じて接続・切断ができれば、それぞれが個別に開発・運用がしやすくなる。この際に、基幹システムが従来のままであっても、データ活用基盤や API 等、連携を仲介する機能を用意すれば、疎結合な形で連携することも可能である。

以上のように、アプリケーション構造は従来のシステムのものとは異なっている。より効率的、且つ少ないコード量で開発ができる最新の開発言語を活用し、アプリケーション開発・運用の俊敏性を高めることを狙ったものである。また、より俊敏なサービス提供・更新に取り組むためには、フェーズごとでの機能確認や再設計となる場合もあるが、MVP (Minimum Viable Product) を基調に俊敏に機能を追加できるようにすることも考慮すべきである。MVP とは、「ユーザーへのサービス提供における、最小限の機能を備えたプロダクト」のことであり、サービスを開発する際に、100%完成品をリリースするのではなく、最小限+αの機能を提供しユーザーの反応を見ながら改善していくための、サービス開発のアプローチである。

DX 時代のビジネスは不確定で不明瞭な領域を対象にしていかなければならない。こうした領域に対して最初から莫大な投資をすることはリスクも大きくなりがちである。しかし MVP を目指すのであれば最初から大きなリソースを動かさずに済むことになる。なぜならまたいち早くユーザーやマーケットに機能を提供でき、反応を知ることができる。このため大きな機能群を実装してしまった後から修正するのとは異なり、比較的軌道修正が容易であるため、結果的には設計や実装にかかるコストが無駄になりにくいからだ。

この要件を満たすアーキテクチャに「マイクロサービスアーキテクチャ」がある。たとえマイクロサービス化まで踏み込んでいなくとも、DX の実践においては、日次等、短い期間でリ

リリース可能な小規模・軽量なアプリ開発は意識しておくことが望ましい。マイクロサービスアーキテクチャの詳細については、第4章で述べる。

② クラウドのような「拡張（容易）性」、システム環境の立上げ/停止を俊敏に行える弾力的な基盤を備えている

変化対応力のあるシステム基盤に必要な要素として、拡張が容易であることと、環境構築やその停止が容易かつ俊敏であることが挙げられる。現在のところ、これらの要素を満たす最も有効な手段は、クラウドの活用であると考えられる。

クラウドコンピューティングは、共用の構成可能なコンピューティングリソース（ネットワーク、サーバ、ストレージ、アプリケーション、サービス）の集積に、どこからでも、簡便に、必要に応じて、ネットワーク経由でアクセスすることを可能とするモデルであり、最小限の利用手続きまたはサービスプロバイダとのやりとりで速やかに割当てられ、提供されるものである²⁰。

クラウドを活用するメリットの一つは、俊敏に環境を構築・撤去できることである。クラウド化には特別な機材の導入や物理的なインフラ構築が必要なく、サービスに申し込むだけですぐに利用が可能で、オンプレミスのシステムに比べシステム構築が比較的容易であり、初期投資コストを大幅に抑えることが可能である。顧客向けのモバイルアプリ等もアジャイルにリリースしていく上でも、パブリッククラウドを活用することは有効な手段である。

また、サービスの成長に応じて拡張や再構築が容易であることもその利点である。ビジネスの需要に応じて即座にスケールアウト／スケールインすることが可能であり、小さくビジネスをスタートし、ユーザー数やデータの取扱量が増加するなどの成功の兆しが見えればそのまま拡大することが可能である。逆に、サービスの需要が見込めなければ撤退を選択することが比較的容易になり、DXに求められる「小さく失敗をする」ことが可能になる。

ただし、医療・製薬業界などのデータに対するセキュリティレベルが高い業界やミッションクリティカルなシステムを扱う組織では、必ずしもパブリッククラウドではなく、社内環境にクラウドを構築するプライベートクラウドやオンプレミスを採用している場合もある。また、以前から顧客個人情報を扱っているがゆえに、既存のオンプレミスのリソースが既に高いレベルで構築されており、その活用のためにもクラウドのメイン活用には舵を切らないという選択をとる先進企業も見られる。

このような事情を踏まえ、オンプレミス、プライベートクラウド、パブリッククラウドを

²⁰ 独立行政法人情報処理推進機構「NIST によるクラウドコンピューティングの定義 米国国立標準技術研究所による推奨」, 2011年9月, <https://www.ipa.go.jp/files/000025366.pdf>

組み合わせて、扱うデータや環境に応じて使い分けるハイブリッドクラウドや、用途に応じて複数のクラウドサービスを活用するマルチクラウドも検討の余地がある。マルチクラウドとは、ハイブリッドクラウドを包含した 3 種類以上のクラウドサービスを、1 企業が用途に応じて使い分けて利用している構成を意味する。ここでは必ずしもマルチクラウド環境が最適であり、推奨をするわけではない。しかしながらオンプレミスや様々なクラウドには、それぞれ特性やサービスの機能差異などがある。このため、ビジネスの実装上、様々なクラウドから実装要件を満たし、かつ最適なものを選択することになる。この結果がマルチクラウド化の傾向を示していると考えられる。事実、グローバルに活躍する企業 433 社に対するクラウド活用戦略調査アンケートでは 8 割ものエンタープライズ企業が「マルチクラウドを活用している、または今後活用していく戦略である」と回答している²¹。

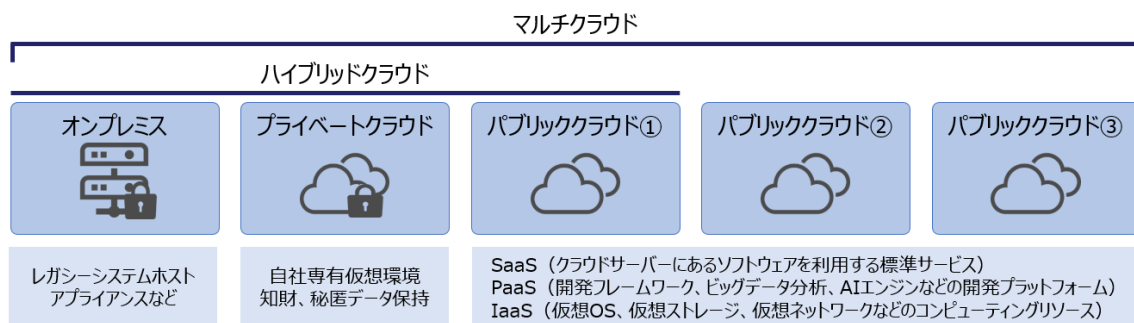


図 3.3.1 マルチクラウドのイメージ

このマルチクラウドの潮流は、クラウドサービス全体の成熟に伴って、企業の選択肢が増加したことにより、それぞれの事業の目的に沿ったクラウドサービスを適材適所で利用できるようになったことに起因する。1 つの組織の中でも、それぞれの目的に応じた選択が可能であり、例えば、ミッションクリティカルなシステムをプライベートクラウドに、顧客向けのコンテンツ配信基盤をパブリッククラウド①に、またビッグデータ処理をパブリッククラウド②にといったように、各々のクラウドサービスが得意とする分野の機能を選択し、組み合わせることで、コスト効果の高い IT システム体系を主体的に実現する。

企業の DX の取り組みにおいては、既存のシステムを抱えながら、新規にシステムやサービス・アプリケーションを構築することが求められ、時間的、資金的制約がある中で、すべてを新たに刷新するのは現実的には難しい。そのため、ハイブリッドクラウドを含む、マルチクラウドを前提に、オンプレミス、プライベートクラウド、パブリッククラウドなど、さまざまな環境に対応できる柔軟性を備えた API 中心のシステムを整備することが理想的であ

²¹ 鳥谷部昭寛「デジタル時代のクラウド活用戦略 マルチクラウド化の潮流」, 2017 年 5 月, <https://www.nri.com/-/media/Corporate/jp/Files/PDF/knowledge/publication/chitekishisan/2017/05/cs20170504.pdf?la=ja-JP&hash=42F325CF6BEBF26FA7DFD38B86FE1A01A4E7DFB7>

る。適材適所にサービスを取り入れることが可能であり、それが常に最適かどうかを判断し、変化に追従していけるような対応が求められている。

3.4. DX を実現する IT システムのあるべき姿の全体像

本章でここまで述べてきた「社会最適」「データ活用」「スピード・アジリティ」の DX を推進するための IT システムの重要な 3 つの要件について、これを備えるようなシステムのあり方を考えると、一般的には図 3.4.1 のように表現できる。

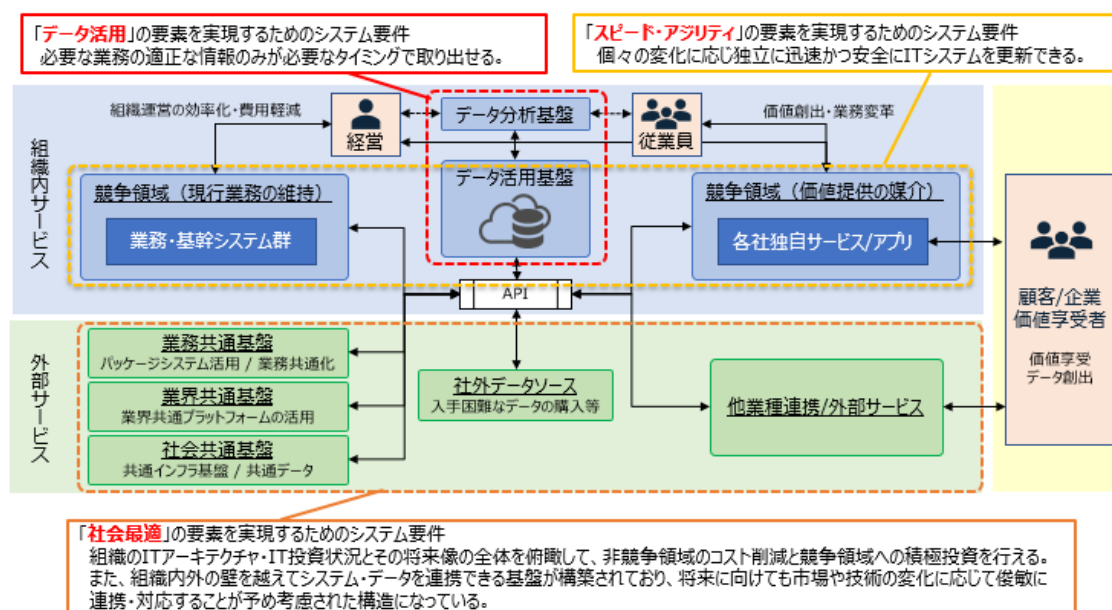


図 3.4.1 あるべき IT システムの要件

ここにおいて、社内の IT システムは、組織内で独自に構成したシステムと外部のプラットフォームを活用したシステムの双方の組み合わせとして構成される。組織内サービスは、各企業の競争領域として、独自の価値を生み出すための IT システム群のみとする。それに対して、各企業の社内リソースを競争領域に注力することを目的として、非競争領域の低コスト化・効率化や、競争領域をさらに強化する目的において他社が提供するサービス等を有効活用するための IT システム群を外部サービスとして調達する。

あるべき IT システムに求められる 3 つの要件について、このあるべき IT システムの要件との関係性を以下に説明する。

① 社会最適

「社会最適」の視点は、図 3.4.1 において、外部共通基盤の利活用による社内業務システムの維持と保守コストの削減、および社外データの活用・連携や、顧客接点を増やしたり、新しい価値を創出するための他業種との連携をしたりすることを踏まえた外部サービス活用という形で体现されている。「社会最適」の視点で組織内外の IT システム全体のあり方を見直すことで、着実にデジタルを前提としたビジネスを前進させる。まずは、組織全体のそれぞれの IT システムの位置づけを分類したうえで、自社がコストをかけて強化すべき競争領域と、外部の有用なリソ

ースの活用も視野に入れた協調領域 / 非競争領域を分けて考えることで、デジタル活用を企業活動の当たり前のものとして、変革を推し進める体制を整えるべきである。

② データ活用

「データ活用」の視点は、図 3.4.1 において、データ活用基盤を中心に据えて、データ中心の経営判断に活用したり、新たな競争領域のサービス開発を進めたり、柔軟性の高いデータ資源活用が可能になる形で体现されている。デジタルエンタープライズにおけるビジネスの意思決定ではデータが大きな役割を担う。市場動向に合わせて前例のない課題に取り組むにあたり、ビジネスのリアルタイムなデータをよりの確に参照できるよう整備し、データを基軸とした判断根拠をもとに意思決定を行っていくような、データドリブン企業となることが求められる。これによって経営判断のスピードアップと精度向上を実現し、より素早く高付加価値化なサービスを提供することを目指す。また先に述べたように、データの取得、分析、活用の領域が広がったことにより、データ活用は既存ビジネスのバリューアップだけでなく、企業の新規ビジネスの創出にも貢献している。必要な業務の適正な情報のみが必要なタイミングで取り出せるような仕組みを構築するために、データ活用に特化した基盤を中心に据える仕組みが有効である。

③ スピード・アジリティ

「スピード・アジリティ」の視点は、図 3.4.1 において、組織内の個々のシステムやアプリケーションが、社会環境の変化に応じて迅速かつ柔軟に変更できる仕組みという形で体现されている。市場や社会の環境変化に対して、サービスや機能を革新し俊敏にデリバリーできるようにするためには、IT システム全体の「スピード・アジリティ」要素を予め考慮しておく必要がある。なぜなら、市場や社会の要請に対して俊敏に対応し、新規に素早く構築したアプリケーションやシステムが、外部接続の汎用性が高く、更新の難易度が低い仕様であっても、運用の際の既存の IT システムとの連携がうまくいかなかったり、連携のための手作業が増え、業務効率が下がったりするケースが想定されるからである。IT システムの構想・設計・開発・運用のそれぞれのフェーズで、IT システムやアプリケーションの個々の変化に応じ独立・俊敏・安全に更新できるような仕組みを、あらかじめ設計・構築し、運用し続けられる準備しておくことが求められる。

以上の 3 つの要件を兼ね備える IT システムに適うものを、DX の実現のために必要なあるべき IT システムとするものである。

第4章 あるべき IT システムとそれを実現する技術要素

第3章では、DXを実現するための要件の観点から、あるべき IT システムの姿を描いた。

本章では、このあるべき IT システムを実現する際にいかなる技術要素を用いるべきかという観点から、鍵となる技術要素について説明する。

4.1. DX を実現するための技術要素群の全体像

4.1.1. あるべき IT システムを実現する技術要素群の全体構成

第3章で述べたあるべき IT システムの姿を、現在一般的な、また先行事例の調査において有効性が明らかになった技術要素を用いて、既存の IT システムを有する企業が実現するとすれば、おおよそ図 4.1.1 のようになると思われる。

このあるべき IT システムを実現する技術要素群に対して、「スサノオ・フレームワーク」という名前を付け、以降、この「スサノオ・フレームワーク」²²の考え方に沿って、説明していく。

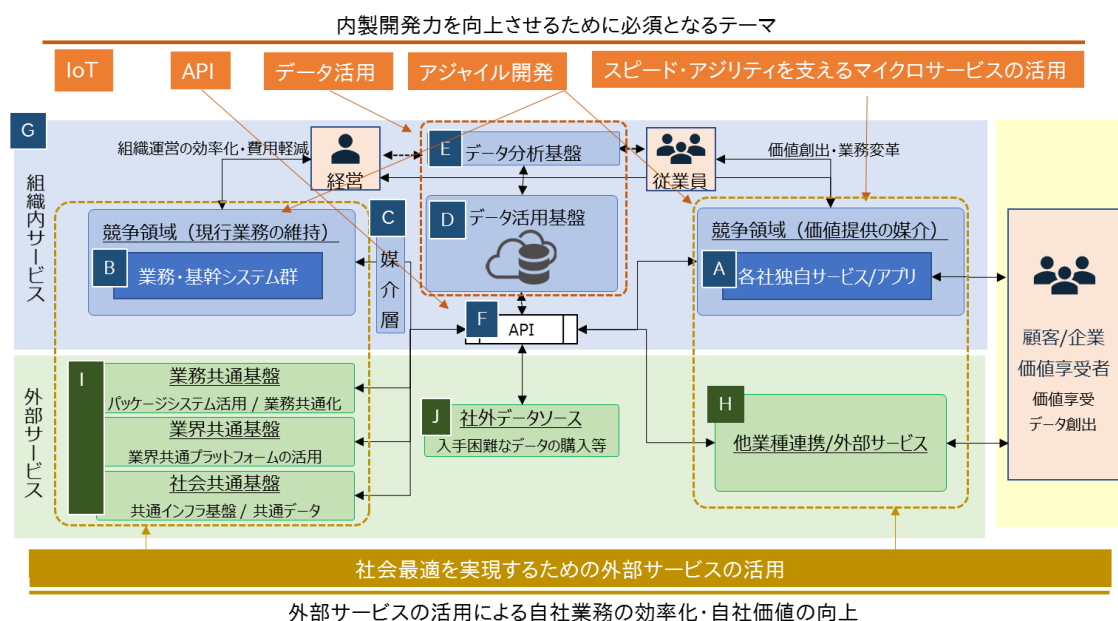


図 4.1.1 あるべき IT システムを実現する技術要素群「スサノオ・フレームワーク」

あるべき IT システムは、各企業の競争領域として、変化に強い独自アプリケーションやそこ

²² モノリシックなシステムであるヤマタノオロチを一つ一つ切り離して、使える部分は形を変えて再生させることで、害となっていた存在を、価値のある存在に変化させるという想いをこめている。

に生かすデータの活用を行う基盤など自社の独自性を強化するための「組織内サービス」と、各企業の社内リソースを競争領域に注力するために、外部のサービスや共通の基盤など、社外のリソースを活用する「外部サービス」を組み合わせた構成となっている。

あるべき IT システムの個々の要素は、社内外の様々なサブシステムと連携するため、API(F)を基本のインタフェースとしている。顧客や市場の変化に耐えうるため、**競争領域の独自アプリケーション(A)**を、顧客や企業と接点を持つアプリケーションとしている。社内の現行業務を維持するとともに、**競争領域の独自アプリケーション(A)**と連携する形で、**現行システム(B)**をおいている。**現行システム(B)**を再構築する際には、移行を助けるため、**媒介層(C)**が他の IT システムとの連携を仲介する。また、**媒介層(C)**は、**現行システム(B)**が他の IT システムと連携する役割も担う。そうした、社内の IT システムや社外から取得するデータを、集約し、蓄積して、分析するため、**データ活用基盤(D)**・**データ分析基盤(E)**をデータ活用のための基盤としている。以上の組織内のサービスについて、システムリソースの柔軟な変更などを可能とするため、**システム基盤(G)**を稼働環境としている。

顧客や企業との設定を持つ**競争領域の独自アプリケーション(A)**が提供する機能を強化するため、特化した個別の機能は、**外部サービス(H)**と連携する。また、非競争領域となる業務は、外部が提供する環境を利用するため、**外部共通基盤(I)**と連携する。企業内にあるデータと、組み合わせることで、整理や分析に活用するため、**社外データソース(J)**を取り込むことも可能としている。

4. 1. 2. あるべき IT システムを実現する技術要素群内の「組織内サービス」

の各要素

「組織内サービス」のそれぞれの要素について、以下で説明する。

(1) 競争領域の独自アプリケーション(A)

競争領域の独自アプリケーション(A)は、BtoB/BtoC を含めた顧客/企業に対して価値を提供する役割を担う IT システムを表す。例えば、顧客のモバイル環境へ、予約の情報や混雑状況など、顧客がその場で必要とする情報をタイムリーに提供する IT システムなどがそれにあたる。

ここは、スピード・アジリティを備え、顧客やそれを取り巻く社会、業界などの変化に対して、追従し、継続的に価値を提供することが必要となる。そのため、変化に合わせて、時間をかけずに、柔軟に機能変更できるアプリケーションの形態をとる必要がある。

つまり、**競争領域の独自アプリケーション(A)**はその企業の独自のアプリケーションとして構築し、短い期間で変更を加え続けていく対象となる。

そうした性質を備えた**競争領域の独自アプリケーション(A)**を構築するには、「マイクロサービスアーキテクチャ」を採用することが適している。「マイクロサービスアーキテクチャ」

では、各機能を疎結合の形で構成し、変化に強いアーキテクチャとして、実用化の事例も出てきているものとなる。

また、**競争領域の独自アプリケーション(A)**を構築するプロセスとしては、「アジャイル開発」手法を採用することが適している。「アジャイル開発」は、アプリケーションに対する要求仕様の変更に柔軟に対応できる手法として、活用されることが増えてきている。

(2) 現行システム(B)・媒介層(C)

現状、独自アプリケーションとして構築した IT システムを保持している場合も、少なくないと想定されるが、**現行システム(B)**は、そうした現行システムを表す。例えば、企業内で利用される在庫管理や調達管理などの IT システムがそれにあたる。

ここは、現在の業務を維持していくためのシステムで、**競争領域の独自アプリケーション(A)**のような頻度で変更が発生せず、それほど対応のスピードを必要とされない IT システムが該当する。現行システムのまま、運用を継続することで、**競争領域の独自アプリケーション(A)**が提供する価値に、影響しないことが前提となる。

ただし、そうした現行システムも、企業内の他の IT システムと情報連携/集約のため**競争領域の独自アプリケーション(A)**や**データ活用基盤(D)**との情報の受渡は必要となり、**媒介層(C)**を間にはさむことで、他との連携手段を確保する。

また、**現行システム(B)**を再構築する際には、**媒介層(C)**が他システムからの接続を仲介し、**現行システム(B)**の移行前、移行後の IT システムへの振り分けを行い、移行期間における利便性の確保を行う。

(3) データ活用基盤(D)・データ分析基盤(E)

データ活用基盤(D)・データ分析基盤(E)は、企業内外からデータを集め、利活用できるようにする環境を表す。例えば、データウェアハウスにデータを集約し、分析ツールを利用して、それらのデータにアクセスするといった IT システムがそれにあたる。

ビジネスの継続的な発展のため、データを利活用し、そこから得られた示唆をビジネスに生かしていくことは、重要な要素となる。そのために、企業内の各 IT システムに散在しているデータを集約し、それらを統合して参照することができるように品質を整え、必要な人が必要な時に利用することを可能にする環境が必要となる。その点で、**データ活用基盤(D)**に、各 IT システムから収集したデータを、統合、整理、蓄積し、**データ分析基盤(E)**から、それらのデータに対して、企業内の人がアクセスし、参照、分析する環境を用意することが欠かせないものとなる。

(4) API(F)

API(F)は、**競争領域の独自アプリケーション(A)・現行システム(B)・データ活用基盤(D)**の各システムが他のシステムに依存せず、疎結合の状態連携するためのインタフェースとなる。

競争領域の独自アプリケーション(A)と**現行システム(B)**の間では、業務間での情報連携が必要となるが、それぞれの独立性が低ければ、一方に対する仕様の変更が他方へも影響する

ケースが増えてくる。それにより、それぞれの IT システムに対する頻繁な変更や俊敏な対応が、難しくなり、スピード・アジリティの要素を欠くこととなる。また、**競争領域の独自アプリケーション(A)**と**データ活用基盤(D)**の間も、同じように、密結合された構成となっていれば、それぞれの変更に対する制約が発生することにつながる。

そうした状況を避け、各 IT システムの間の独立性と接続性を確保するため、**API(F)**/ **媒介層(C)**を介した連携を可能とする構成をとる。

(5) システム基盤(G)

システム基盤(G)は、組織内の各サービスが稼働する基盤を表している。

競争領域の独自アプリケーション(A)について、顧客が直接利用するような IT システムの場合、一時的に利用負荷が大きく変化するようなことも想定される。そうした際に、必要なシステムリソースを割り当て、不要となれば開放するといったことを、即座に柔軟に行うことが、IT システムを運用する上でのスピード・アジリティにも影響してくる。また、**データ活用基盤(D)**において、データの活用が進めば、蓄積するデータの量や種類は増えることが多く、稼働する環境により多くのリソースを必要とすることになる。

そうした状況へよりスムーズに対応するためには、基盤としては、クラウドを活用することが適している。それにより、拡張容易性の確保、運用負荷の軽減など変化への対応を実現できる。

4. 1. 3. あるべき IT システムを実現する技術要素群内の「外部サービス」の各要素

次に、「外部サービス」の各要素について、説明する。

(1) 外部サービス(H)

外部サービス(H)は、他の企業が提供するサービスを表す。

競争領域の独自アプリケーション(A)の優位性は独自アプリケーションにより実現されるが、そのアプリケーションの機能を補完するものとして外部が提供するサービスにマッチしたものがあれば、それを活用することが有効となる。新規に**競争領域の独自アプリケーション(A)**を構築する場合、既に用意された機能が外部から提供されていれば、それを利用することで、顧客へより早期に提供することが可能になる。また、運用する上でも、自社で対応するよりコスト面で有利であれば、外部サービスの利用によるメリットを生かすことができる。

外部サービス(H)は **API(F)**を通じて**競争領域の独自アプリケーション(A)**と連携し、**競争領域の独自アプリケーション(A)**と**外部サービス(H)**が一体となって、顧客へ価値を提供する役割を担う。

(2) 外部共通基盤(I)

外部共通基盤(I)は、外部サービスの中で非競争領域となる機能を担うシステムで、外部が提供するものを表す。

外部の提供の仕方としては、大きく3種類あり、一部の業務に関するもの(=業務共通基盤)、業界としての共通業務に関するもの(=業界共通プラットフォーム)、社会全体での共有する機能に関するもの(=社会共通基盤)となる。例えば、一部の業務に関するもの(=業務共通基盤)としては、グループウェアや会計管理パッケージなどがそれにあたる。業界としての共通業務に関するもの(=業界共通プラットフォーム)としては、水道事業の共通プラットフォームなどがそれにあたる。社会全体での共有する機能に関するもの(=社会共通基盤)としては、gBizID²³などがそれにあたる。

外部が提供する共通基盤を活用することで、現行業務の稼働、維持、変更に必要なリソースを抑え、**競争領域の独自アプリケーション(A)**によりリソースを投入し、競争力の確保/維持につなげることができる。

(3) 社外データソース(J)

社外データソース(J)は、外部が提供するデータとなり、**データ活用基盤(D)**に取り込むことで、企業内のデータと組み合わせ、整理や分析に活用する。例えば、外部が管理し、提供している地図情報などが、それにあたる。

外部のデータを軸として、企業内に蓄積したデータを整理することで、新たな視座や観点が得られることも少なくない。そこで得られたものを、ビジネスへフィードバックし、顧客への新たな価値を提供することへつなげていくことが、ビジネスの発展に寄与することになる。

4.1.4. 「組織内サービス」の各要素におけるセキュリティの考え方

「組織内サービス」の各要素の設計・実装にあたり必要となるセキュリティの考え方について、以下に説明する。

(1) 対策は多層的に行うことを認識し、責任分担を明確化する

現下のセキュリティ脅威に対応するためには、データ、アプリケーション、OS、ネットワーク、ハードウェア等の各レイヤーにわたる多層的な対策が必要であり、どのレイヤーについて誰が責任を持つのかを明確にしなければならない。例えば独自アプリケーションをクラウド上に構築する場合(図 4.1.1 の(A) (D) (E)等を想定)、クラウド事業者との責任分担が発生する。ハードウェアやネットワークの対策はクラウド事業者の責任となる一方、アプリケーションの脆弱性対策等は自組織の責任となる。個人情報や営業秘密情報をクラウド

²³ デジタル庁が提供する法人・個人事業主向け共通認証システム。 <https://gbiz-id.go.jp/top/>

ド上に保管する場合も、暗号化・廃棄は自組織の責任である。また自組織内でシステムを構築・運用する場合（(B) (C) (D) (E) 等を想定）は、開発委託・運用委託事業者とのセキュリティに関する責任の明確化が重要である。一方で(C)のオンプレミス環境については、ネットワーク・ハードウェア等基盤システムの権限管理も重要である。最小特権の原則に基づき、重要度の高いシステム基盤へのアクセス権は必要最小限にすることが望ましい。

さらに、外部サービス（(H) (I) (J) を想定）利用において、自組織のセキュリティに関する責任を理解することも重要である。例えば、外部サービス利用時の ID・認証情報管理や、外部公開するオープンな API ((F) を想定) の脆弱性解消は自組織の責任である。また、現在連携している、及び今後連携を検討する外部サービスのセキュリティ機能が自組織のセキュリティポリシーに適合していることを確認することも必要である。

上記に基づき、スサノオ・フレームワークのセキュリティ責任境界と各レイヤーの対策例を図 4.1.2 に示す。赤い実線が責任境界、赤字が対策である。責任境界内においても、委託先等との責任分担があり、その最終責任は自組織にあることに注意されたい。

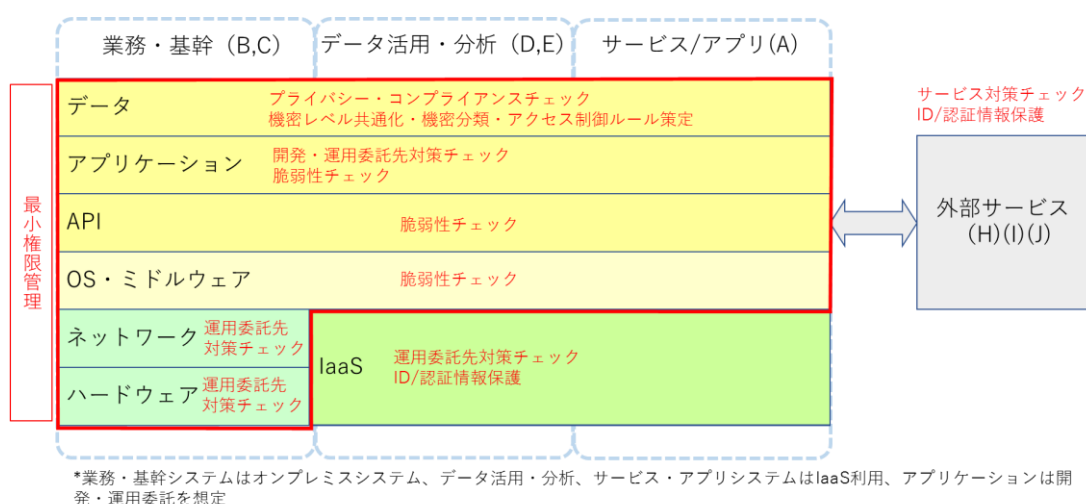


図 4.1.2 スサノオ・フレームワークのセキュリティ責任境界とレイヤーごとの対策

(2) 守るべき資産(データとシステム)を明確化し、資産の重要度に基づいたセキュリティ対策・データ共有を実施する

重要な資産を特定すること、図 4.1.1 でいえば (A) (B) (D) で利用するデータの機密レベルを決定すること、機密レベルの高いデータを処理するシステム（業務）を特定することはセキュリティリスク管理の基本である。重要な資産を特定したら、その重要度に基づいたセキュリティ対策（アクセス制御、暗号化、監視等）を実施することになる。組織全体で同じ強さの対策を実施するため、機密レベルやアクセス制御ルールは共通化することが望ましい。また (1) で述べた通り、外部サービスの利用においては、当該外部サービスのセキュリティ機能が自組織で担保したい強さを満たすことを確認する必要がある。

(3) 開発においては設計時からセキュリティ機能の作りこみを行い、開発環境もセキュアに

保つ

設計時におけるセキュリティ機能の作りこみは「セキュリティ・バイ・デザイン」とも呼ばれる。特に API による連携を志向する場合、情報漏えいやサイバー攻撃のリスクを最小化するため、API を介したデータ共有は必要最小限にすることが重要である。例えば、基幹業務 (A) の機微データに関して媒介層 (C) でフィルタリングや整形を行い、(B) (D) に提供する範囲を必要最小限とする。また (A) (B) (D) (E) と外部サービス (H) (I) (J) が連携する場合、提供するデータも外部サービスが求める範囲で最小限とし、必要に応じて整形する。外部公開するオープンな API と自組織のみで利用する API 等を切り分けることも重要である。これらは設計段階で吟味しておく必要がある。さらに開発においては、セキュア開発やセキュアコーディングに関するガイドラインに準拠する、等の脆弱性を作りこまない施策が重要である。

また、開発環境は図 4.1.1 には登場しないが、セキュアかつ迅速なソフトウェア開発を行うため、セキュリティチェックシートの拡充や一部自動化などの開発者支援が重要である。また、開発環境をセキュアに保ち、開発ツールやライブラリに脆弱性等が発見されたら素早く対応することも求められる。

(4) データはセキュリティに加え、プライバシー・コンプライアンスルールに基づいた管理を行う

データ活用基盤 (D)、データ分析基盤 (E) においては、機密性保護の観点からのデータ暗号化・アクセス制御等のセキュリティ要件に加え、データ収集・利活用や分析を円滑に行うため、プライバシーやコンプライアンスルールに基づいた管理が必要である。特に個人情報、あるいは個人の行動や業務執行状況等に関するデータを収集、分析し、結果を利活用する、さらにその業務を外部と連携して行う、等においては、誰が何の目的でそのデータにアクセスするのか、その利用にプライバシーやコンプライアンス上問題はないか、をチェックし、必要な場合は本人同意を得る、あるいは (H) (I) (J) にデータを提供する場合に匿名化・仮名化等の処理を行う、開示範囲を限定する、等で分析・利活用が問題なく行われることを担保しなければならない。

なお、プライバシー・コンプライアンスについても、設計時から、専任のチームあるいは専門家を含めた検討を行うことが重要である。例えば、あるサービスの利用データを AI で学習してサービスを向上させたい、また将来的には別なサービスと統合したい、という場合、データ利用に関する利用者との合意やデータ処理はどうすればよいか、サービスやデータ統合によるプライバシー上の問題は起こらないか、等について、今後の利用形態を想定して十分に検討することが必要である。

4.1.5. あるべき IT システムとクラウドの関連

本項では、「スサノオ・フレームワーク」とクラウドとの関係について、説明する。

3章において、「スピード・アジリティ」を実現するための要素の1つとして、クラウド活用に関して述べている通り、システム基盤としてクラウド環境を利用することは、現実的な選択肢となってきた。

クラウドは、NIST の定義²⁴によれば、代表的な形態として、IaaS(Infrastructure as a Service)、PaaS(Platform as a Service)、SaaS(Software as a Service)に、分類される。

オンプレミスも含め、先のクラウドの形態は、以下のように、図示することができるだろう。

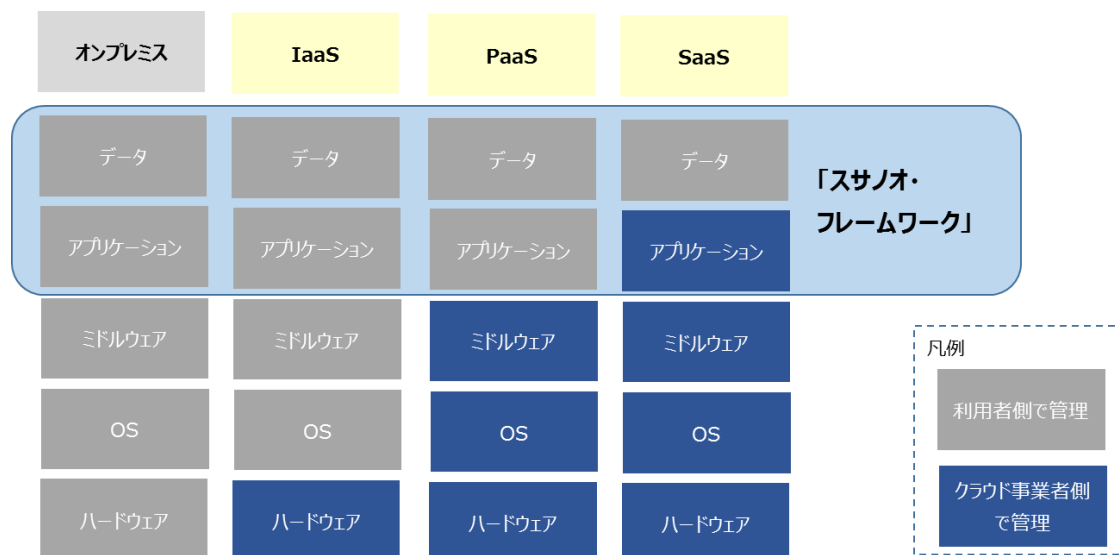


図 4.1.3 クラウドと「スサノオ・フレームワーク」の関連

それに対し、「スサノオ・フレームワーク」で表している姿は、上記の枠線で囲まれた領域に該当する。OS やミドルウェアの上で、稼働するアプリケーションとそこで取り扱うデータが、「スサノオ・フレームワーク」が担当する領域となる。それ以外の領域については、「スサノオ・フレームワーク」としては対象外の領域となり、利用する形態に合わせ、必要となる基盤を選択する。

こうした、クラウドをシステム基盤として利用するメリットとしては、IT システムを稼働させるためのリソースに対する柔軟性や拡張性が高い点などが挙げられる。また、クラウド側から様々なサービスが提供されている。これらのサービスを活用することで開発、運用管理の省力化を行うことも可能となる。

例えば、**競争領域の独自アプリケーション(A)**において、クラウドでは、システムリソースを柔軟にコントロールすることができるため処理量の変化などに対応することができ、アプリケーションの実行環境としても有用となる。また、クラウド側から IT システムの開発や運用に利用できるサービスが提供されていることも多く、それらを使い、開発期間の短縮や運用負荷の軽減などを、実現することもできる。

データ活用基盤(D)においても、必要なタイミングでデータの格納のための領域の拡張/縮小が可能であるので、対象となるデータの増減などにも、スムーズに対応することができる。

²⁴ <https://www.ipa.go.jp/files/000025366.pdf>

一方で、特定のクラウドの実装に依存したアプリケーションの作り方をしてしまうと、そのクラウドサービスにロックインされてしまい、他のサービスへの乗り換えに大きなコストと時間が必要となり、他のサービスが選択肢となりえない状況になることもあるため、注意が必要である。

クラウドのメリットを活かしつつ、クラウド側で活用すべき機能と「スサノオ・フレームワーク」側で担当すべき機能を見極めながら、活用していくことが必要と考えられる。

4.1.6. あるべき IT システムと IoT の関連

企業における業務を進める上で、また日々の生活をする上で、IoT (Internet of Things :モノのインターネット) によって、様々なものがつながることで、現場やそこで起きていることをデータとして吸い上げ、それらのデータを活用することが可能となる。

そこで得られたデータを集約し、分析することで、現場での作業の効率化や高度化、新たな価値を生むサービスの創出などに生かすこともできるようになってきている。このように IoT の活用により、生産性の向上やサービスの変革に生かすことは、本手引書で解説している DX とも深い関係がある。

本項では、そうした IoT に関し、「スサノオ・フレームワーク」との関係について、説明する。

そうした IoT と関連する要素についての階層関係を表すモデルとして、IEC62264 で示された資源の階層モデル(Equipment hierarchy model)を挙げることができるだろう。(なお、インダストリー4.0 の参照モデル RAMI4.0²⁵において、一部拡張され、階層レベル(Hierarchy Levels)として示されている。)



図 4.1.4 RAMI4.0 の階層レベル

その階層レベルを元にする、「スサノオ・フレームワーク」は、以下の図 4.1.5 のように位

²⁵ https://www.plattform-i40.de/IP/Redaktion/EN/Downloads/Publikation/rami40-an-introduction.pdf?__blob=publicationFile&v=3

置付けることができるだろう。そして、「スサノオ・フレームワーク」の中の**データ活用基盤**が、**IoT**との接点となって、「スサノオ・フレームワーク」と **IoT** の間で、データ連携が行われる関係となる。

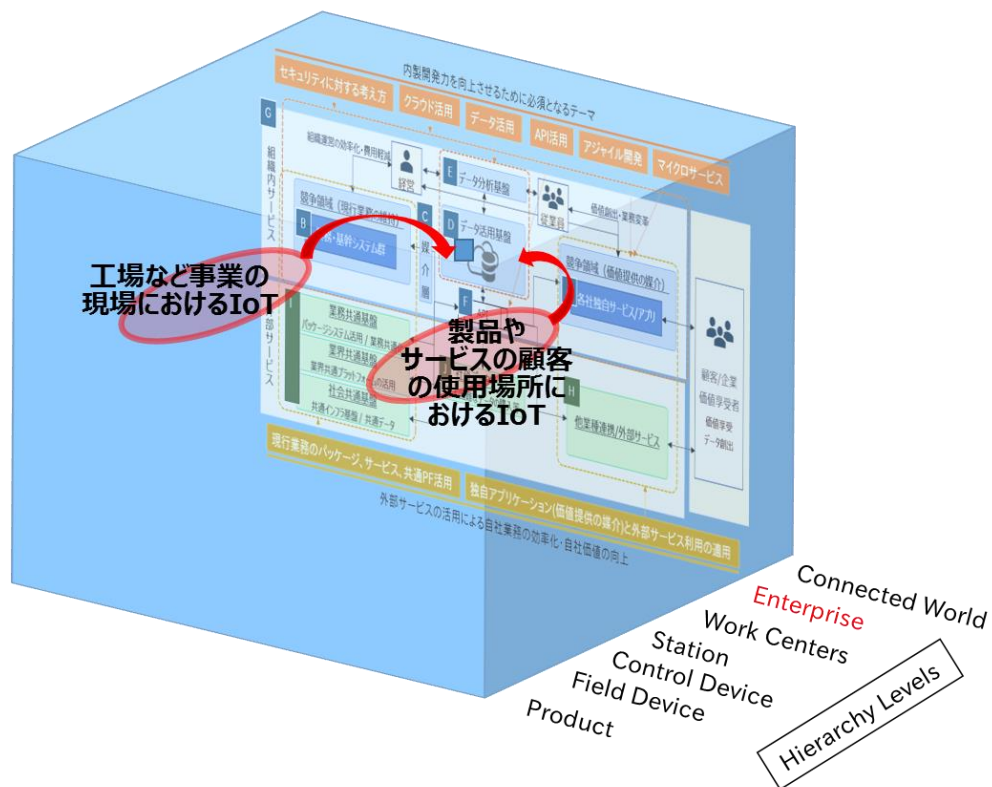


図 4.1.5 「スサノオ・フレームワーク」の位置付けとIoTとの関連

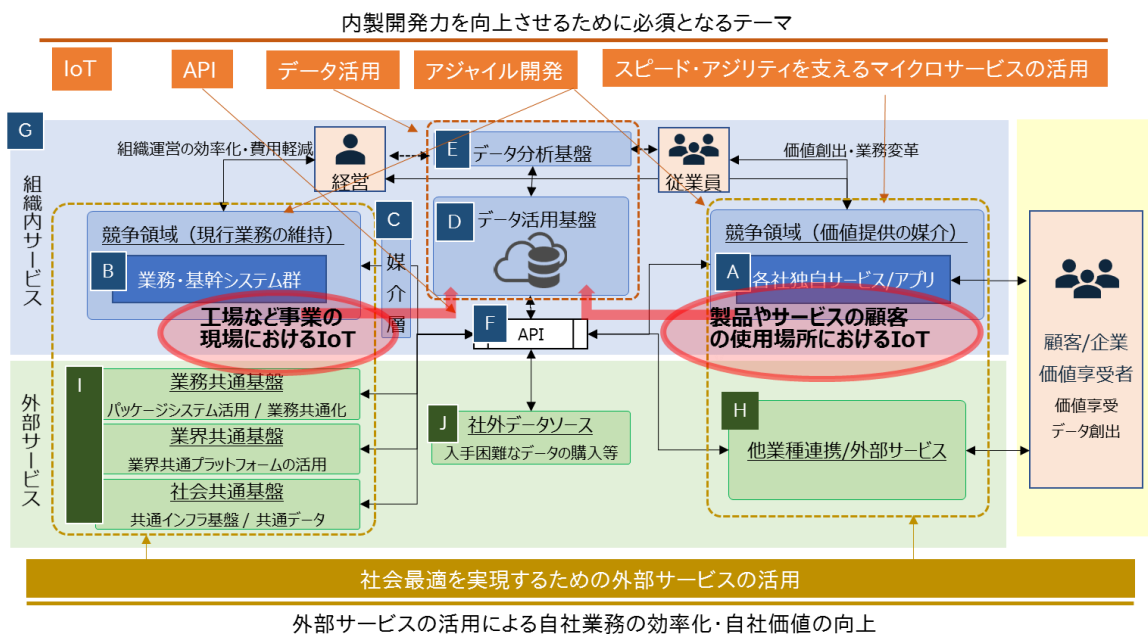


図 4.1.6 「スサノオ・フレームワーク」とIoTとの関連

(図 4.1.5 を「スサノオ・フレームワーク」の正面から見た図)

具体的には、リアルシステムの情報(例えば、製品を生産する工場の機械設備稼働状況データ、製品/サービスの顧客からの使用データ、等)をセンサー(**Product**)でデジタル化し、製造ステーション(**Station**)でデータ化し、工場(**Work Centers**)の単位で情報化し、それをデータ活用基盤に置いて分析することで知識化する。そして、それを逆の流れでリアルシステムに反映させるといった形で、連携する関係となる。

ただし、リアルシステムからのデータの連携について、IoT の各層とのつながり方は、求められるレイテンシーにより、その方式や頻度などが異なってくる。データ連携する層が増えるに従い、タイムラグを極力小さくするためにレイテンシーへの配慮が必要となる、そのため、連携するデータの頻度を落とす、データの粒度を粗くするといった対応が必要となる。

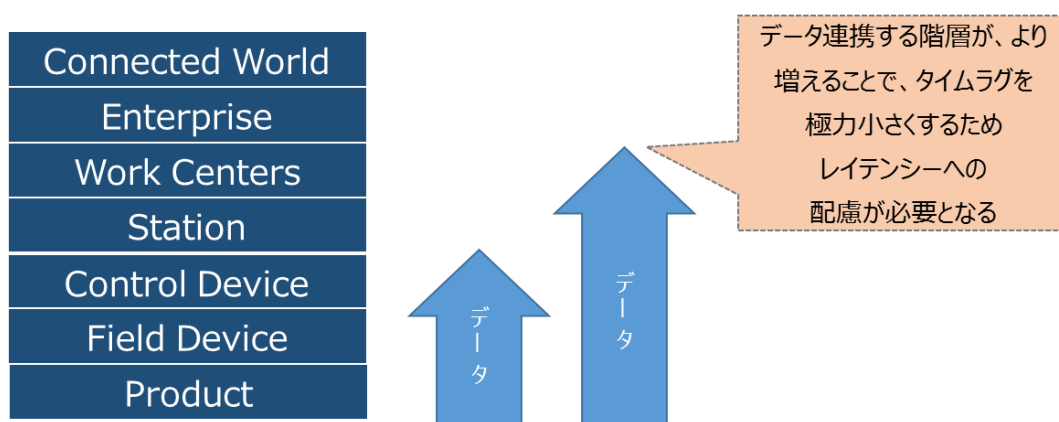


図 4.1.7 レイテンシーと階層の関連

なお、IoT の説明や DX との関連、具体的な活用事例については、「4.6 IoT」にて、解説する。

4.2. アジャイル開発

本項では、図 4.2.1 に示すスサノオ・フレームワークにおける競争領域のシステム構築にアジャイル開発を適用することにより得られる効果や考慮すべき事項について述べる。

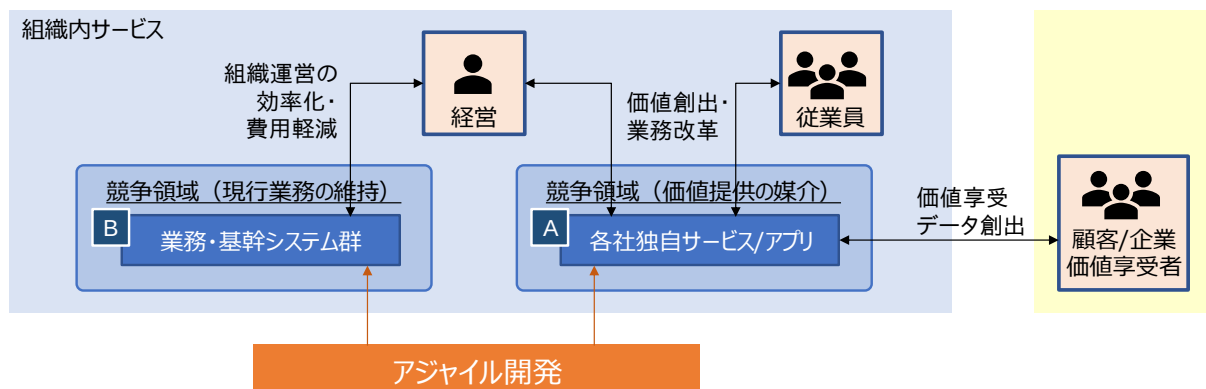


図 4.2.1 アジャイル開発のスサノオ・フレームワークにおける位置づけ

4.2.1. アジャイル開発が必要とされる背景

現代のソフトウェア開発において経営的な関心事としては、以前のような「いかに効率化して人件費等のコストを削減できるか？」から、「いかに新しい価値を創造して競争優位を獲得できるか？」に変わってきている。DX に対応するためにソフトウェア開発も、目はこちらの方に向いている。一方、新しい価値の創造は単純な一本道ではなく試行錯誤を伴うことが多いが、従来主流の予測型のソフトウェア開発手法は当初計画を柔軟に変更することを想定しておらず²⁶、新たな価値を発見しても軌道修正が難しいため試行錯誤に向いていない。この問題の解決策として、適応型²⁷で仮説検証（価値探索）を繰り返すアプローチであるアジャイル開発が有用と考えられている。

具体的なアジャイル開発のイメージを掴んでもらえるよう、最初に中小製造業における成功事例を紹介したい²⁸。A 社は IC カード基盤のプレス加工を得意とする創立 46 年の会社である。

デジタル化のニーズ	製造の作業記録をトレーサビリティのため 10 年間保管する必要があり、当初は倉庫を借りて紙で保管していたが、最近ではタブレット等が普及してきたのでペーパーレス化を検討した。
従来の方法の課題	製造現場にマッチする市販パッケージがないため新規開発を検討したが、過去に現場にノート PC を導入して定着しなかった経験があるため、本当に使えるシステムの開発が難しいことを経験的に知っていた。各社から見積りを取得したが、多くは当時の日報をそのままタブレット化した提案であり、要件定義しても本当に現場で使えるか確信を持てず、契約には至らなかった。
アジャイル開発	会社探しに 1～2 年を費やしたが、自治体の DX 支援事業で知り合った会社から

²⁶ 予測型の場合、工程毎に必要な人材や人数が異なるので要員計画の都合上、スケジュール変更に柔軟に対応することは容易ではない。また請負契約の場合は契約上の制約として成果物を変更できないケースもある

²⁷ 適応型はメンバーを固定して工程を反復するアプローチのため、計画変更に柔軟に対応することが比較的容易

²⁸ 2.4 の事例の(1)と同様の事例。

発の長所	アジャイル開発の提案を貰い、最初は記録だけの小さな機能を約 10 日で作った。現場リーダー3 名でトライアルを始めたが現物に触れながら進めるのが良かった。IC カード方式は費用が高いので QR コードに変更し、カメラが直ぐ起動してスキャンして終了ボタンを押すだけ。ストレスがないことを最優先とし、複雑な機能等は却下した。すぐに結果が出るので現場から要望がよく出るようになった。
効果	ペーパレス化が当初の目的だったが、現場の要望により機能拡張してリアルタイムで「見える化」した。これを 2～3 カ月続けたら生産性が上がることがわかったので外販に踏み切り、50 社以上に導入した。新しい顧客が増えて新しい事業に繋がり、ビジネスが広がっている。

このように A 社は製造日報入力という基幹業務のデジタルオペティマイゼーションと新サービスの創出というデジタルトランスフォーメーションの両方を実現したが、ブレークスルーのきっかけは本当に使えるシステムを開発するためアジャイル開発を選択したことであった。一般論として、本当に使えるシステムの開発が実は難しいことは社外に開発を依頼した多くの人経験していると推察するが、ある調査²⁹によればリリースされたシステム機能のうち約 3 分の 2 はあまり使われていない。使われないシステムが価値を発揮することはないのでムダな開発である。一方、アジャイル開発はトヨタ生産方式（以降、「TPS」と省略。）の「徹底的なムダ排除」という思想を継承しており、必要なモノを、必要な時に、必要な分だけ造る「ジャスト・イン・タイム」と同様に、最短で MVP ³⁰を構築する。A 社の事例にある通り、最初は小さな機能から始めて現物に触れて検証しながら大きくするので、本当に使える（つまり、価値を発揮する）システムを開発できる可能性が高いアプローチと言えるだろう。

4.2.2. アジャイル開発とは何か

IPA は「アジャイル開発とは、ビジネスの価値の最大化に向けて、顧客に価値のあるソフトウェアを早く、継続的に提供するためのアプローチ」としているが³¹、アジャイル開発とは価値観や原則であり、ソフトウェア開発手法であり、新規事業の開発手法であり、組織変革や働き方改革の手段であり、応用範囲が広く有用だが、抽象的なため理解が難しい面がある。そのため本質を理解せずに形式だけ真似て失敗するケースも多いと思われる（4.3.4 にて後述）。そこで本質かつ土台である価値観と原則が生まれた背景と、土台の上に柱を立てて屋根をかけることにより現在の形に至った経緯をアジャイル開発の歴史を振り返ることによって説明したい。

表 4.2.1 アジャイル開発の歴史

年代	主な出来事
1950 年代	1950 年頃よりトヨタ自動車が TPS を製造ラインに段階的に適用 ³² 米国 NACA（NASA の前身）が極超音速機 X-15 の開発に反復型開発を適用 ³³

²⁹ IPA 「IPA/SEC におけるアジャイル開発に関する 4 年間の取組みから分かったこと」 p.18、2012 年 12 月、<https://www.ipa.go.jp/files/000005490.pdf>

³⁰ Minimum Viable Product の略。実用最小限のプロダクト

³¹ IPA 「アジャイル開発の進め方」 p.33、2020 年 2 月、<https://www.ipa.go.jp/files/000065606.pdf>

³² トヨタ自動車「トヨタ自動車 75 年史／トヨタ生産方式」、2012 年 11 月、https://www.toyota.co.jp/jpn/company/history/75years/common/pdf/production_system.pdf

³³ W.D.Kay 「The NASA History Series. FROM ENGINEERING SCIENCE TO BIG SCIENCE. Chapter6 The X-15 Hypersonic Flight Research Program」、2001 年、<https://history.nasa.gov/SP-4219/Chapter6.html>

1960 年代	米国 NASA がマーキュリー計画に反復型開発を適用 ³⁴
1970 年代	米国 IBM や TRW が陸軍や海軍向けソフトウェア開発に反復型開発を適用 ³⁴
1980 年代	竹内弘高、野中郁次郎の論文がハーバード・ビジネス・レビュー誌に発表 ³⁵
1990 年代	1990 年に TPS を一般化・再体系化した、リーン思考の著書が出版 ³⁶ スクラム等の軽量ソフトウェア開発手法 ³⁷ の活用が広がる
2000 年代	2001 年にアジャイルソフトウェア開発宣言により共通の価値観や原則を定義
2010 年代	リーン思考をビジネスに応用したリーン・スタートアップの著書が出版 ³⁸ アジャイル開発の原則を組織改革や働き方改革に取り入れる動きが広がる

先述の通りアジャイル開発の源流の 1 つは TPS であるが、トヨタには「知恵と改善」と「人間性尊重」という価値観がある³⁹。「知恵と改善」とは常に現状に満足することなく、より高い付加価値を求めて知恵を絞り続けることであり、その価値観は製造ラインのandonによる問題の見える化、異常時に設備が自動的に停止して不良品発生を防止する自動化⁴⁰、問題の原因調査及び再発防止のためのカイゼン活動等に反映されている。そして図 4.2.2 に示す通り、後述する「スクラム」の透明性／検査／適応という考え方に継承されている。



図 4.2.2 TPS の「知恵と改善」とスクラムの「透明性／検査／適応」の関係性

また「人間性尊重」は、あらゆるステークホルダーを尊重し、従業員の成長を会社の成果に結びつけるという意味だが、これも後のアジャイルソフトウェア開発宣言やスクラムの価値基準に継承されていると思われる。他に平準化生産、多能工化、品質を工程にてつくり込む考え方⁴¹等、TPS とアジャイル開発には多くの共通点がある。TPS は従来主流だった大量生産方式に対して多品種少量生産を柔軟かつ効率よく行うことを可能にした革新的なシステムなので、新しいソ

³⁴ Craig Larman、Victor R.Basili 「Iterative and Incremental Development: A Brief History」、2003 年 6 月、IEEE Computer. 36

³⁵ Takeuchi Hirotaka、Nonaka Ikujiro 「The New New Product Development Game」 1986 年 1 月、Harvard Business Review

³⁶ James P. Womack、Daniel T Jones、Daniel Roos 「The Machine That Changed the World: The Story of Lean Production」、1990 年 11 月、Harper Perennial

³⁷ 軽量ソフトウェア開発は（計画通りの遂行のために過剰な管理を行うのでなく）俊敏性や柔軟性を重視する開発手法であり、例として RAD、UP、DSDM、Scrum、Crystal Clear、XP、FDD 等が挙げられる

³⁸ Eric Ries 「The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses」、2011 年 9 月、Crown Business

³⁹ トヨタ自動車「トヨタ自動車 75 年史／トヨタウェイの編纂」、2012 年 11 月、https://www.toyota.co.jp/jpn/company/history/75years/text/leaping_forward_as_a_global_corporation/chapter4/section7/item4.html

⁴⁰ ニンベンの付いた自動化。人の働きを機械に置き換える事。

⁴¹ テスト駆動開発における Clean Code that works の考えに近い。

ソフトウェア開発手法を必要とする当時の人々に多くの気づきや影響を与えたものと思われる。

1986年に竹内弘高と野中郁次郎が「The New New Product Development Game」を発表するが³⁵、この論文（以降、「竹内・野中論文」と省略。）では自動車やカメラにおける革新的なモノ作りの事例研究を元に次のような考えを述べている。

新製品開発という速さと柔軟さが求められる場面では、成果物を紙に書き、それを壁越しの別のチームに渡すようなリレーを doing はだめである。様々な専門性を持った人が1つのチームを組み、ラグビーのように開発の最初から最後まで一緒に働くことが求められる。人とチームを重視し、彼らに自律的に動ける環境を与えることでブレイクスルーが起こりやすくなると同時に製品化までの時間が短くなる

図 4.2.3 竹内・野中論文の主旨⁴²

そしてラグビーのようにチーム一体となり開発フェーズを重複させて進めるアプローチを「スクラム」と名付け、これが後述するソフトウェア開発手法「スクラム」が生まれる端緒となった。また「スクラム」チームの特徴として次の6点を挙げており、TPS同様にモノ作りにおける人間（多様性やチームワークによる価値創造等）や知恵（学びと継承等）を重視している。

1. 不安定な状態を保つ
2. プロジェクトチームは自ら自己組織化する
3. 開発フェーズを重複させる
4. 「マルチ学習」
5. 柔らかなマネジメント
6. 学びを組織で共有する

図 4.2.4 「スクラム」チームの特徴⁴²

1990年代に多くの軽量ソフトウェア開発手法が登場するが、その代表的な例が「スクラム」であり、TPSを再体系化・一般化したリーン思考を継承している。図 4.2.5に概要図を示す。

⁴² 平鍋健児、野中郁次郎、及部敬雄「アジャイル開発とスクラム第2版」、2021年4月、翔泳社

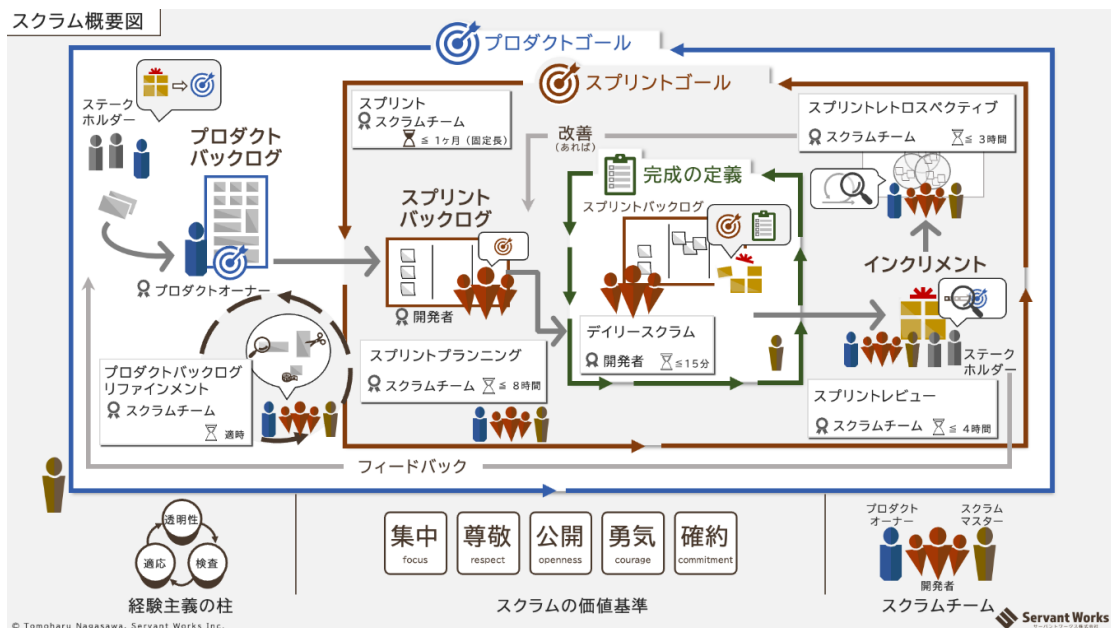


図 4.2.5 スクラム概要図 ⁴³

スクラムとは、複雑な問題に対応する適応型のソリューションを通じて、人々、チーム、組織が価値を生み出すための軽量級フレームワークであり、予測可能性を最適化してリスクを制御するために、反復的で漸進的なアプローチを採用している⁴⁴。説明がやや抽象的なので図 4.2.6 にて補足するが、山岳トレイルを想像してもらいたい。この場合の当初計画は5日間の工程だが、山の天気は変わり易く予測可能な天候は1日先迄のため毎日打合せして翌日どこまで進むか計画する。このとき天気予報や現在位置等の情報を収集して共有し（透明性）、現在の状況を総合的に判断し（検査）、天候が崩れる等の場合は当初計画を変更して臨機応変に対応する（適応）。これを毎日繰り返す（反復）少しずつ前進する（漸進）イメージである。

⁴³ サーバントワークス株式会社「スクラム概要図 2022 年 8 月 8 日版」、2022 年 8 月、https://www.servantworks.co.jp/resources/scrum_overview_figures/

⁴⁴ Ken Schwaber, Jeff Sutherland「スクラムガイド」p.6、2020 年 11 月、<https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-Japanese.pdf>

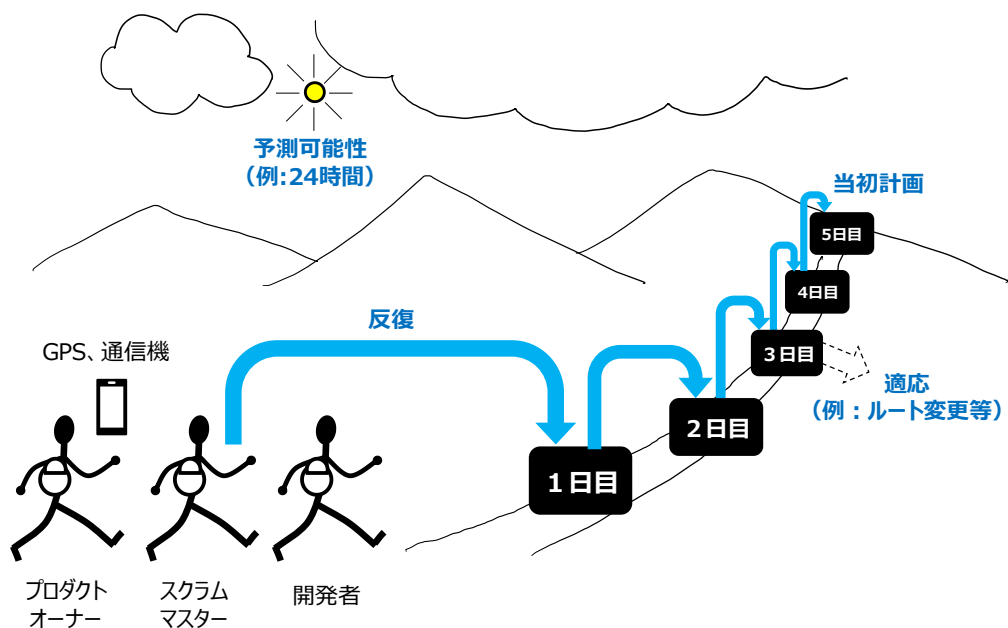


図 4.2.6 スクラムを山岳トレイルに喩えたイメージ

無事にゴールに到着するため重要なのはチームが5つの価値基準（確約、集中、公開、尊敬、勇気）を実践することである。そうしないとチームに信頼関係が構築されず、透明性／検査／適応の3本柱が有効に機能しないからである。スクラムの5つの価値基準を整理した³⁹ものが以下の表であるが、試みにTPSと比較すると、両者の近似性がよくわかる。

スクラムの 価値基準	確約：スクラムチームは、ゴールを達成し、お互いにサポートすることを確約する 集中：スクラムチームは、ゴールに向けて可能な限り進捗できるように、スプリントの作業に集中する 公開：スクラムチームとステークホルダは、作業や課題を公開する 尊敬：スクラムチームのメンバーは、お互いに能力のある独立した個人として尊敬し、一緒に働く人たちからも同じように尊敬される 勇気：スクラムチームのメンバーは、正しいことをする勇気や困難な問題に取り組む勇気を持つ	
トヨタウェイ 2001	知恵と改善	チャレンジ: 夢の実現に向けて、ビジョンを掲げ、勇気と想像力をもって挑戦する
		改善: 常に進化、革新を追求し、絶え間なく改善に取り組む
		現地現物: 現地現物で本質を見極め、素早く合意、決断し、全力で実行する
	人間性尊重	リスペクト: 他を尊重し、誠実に相互理解に努め、お互いの責任を果たす
		チームワーク: 人材を育成し、個の力を結集する

その後、2001年に従来のソフトウェア開発のやり方と異なる手法を実践していた17名のソフトウェア開発者が、それぞれの主義や手法についての議論を行い、「アジャイルソフトウェア開発宣言」として共通の価値や原則を公開した⁴⁵。その価値と原則には、TPSのチャレンジ／改善／現地現物／リスペクト／チームワークを重視する考え方や、竹内・野中論文において立場の異なる人々によるチームワークを重視する考え方と多くの共通点を見出すことが出来る。

⁴⁵ 各々の価値基準については、IPA「アジャイルソフトウェア開発宣言の読みとき方」、2020年2月、<https://www.ipa.go.jp/files/000065601.pdf>を基に展開

アジャイルソフトウェア開発宣言

私たちは、ソフトウェア開発の実践
あるいは実践を手助けをする活動を通じて、
よりよい開発方法を見つけだそうとしている。
この活動を通して、私たちは以下の価値に至った。

プロセスやツールよりも**個人と対話**を、
包括的なドキュメントよりも**動くソフトウェア**を、
契約交渉よりも**顧客との協調**を、
計画に従うことよりも**変化への対応**を、

価値とする。すなわち、左記のことがらに価値があることを
認めながらも、私たちは右記のことがらにより価値をおく。

Kent Beck	James Grenning	Robert C. Martin
Mike Beedle	Jim Highsmith	Steve Mellor
Arie van Bennekum	Andrew Hunt	Ken Schwaber
Alistair Cockburn	Ron Jeffries	Jeff Sutherland
Ward Cunningham	Jon Kern	Dave Thomas
Martin Fowler	Brian Marick	

© 2001, 上記の著者たち
この宣言は、この注意書きも含めた形で全文を含めることを条件に
自由にコピーしてよい。

図 4.2.7 アジャイルソフトウェア開発宣言の価値

2010 年代にアジャイル開発はビジネス分野にも活用を広げるが、その端緒となったのがリーン・スタートアップである⁴⁶。リーン・スタートアップはTPSを一般化・再体系化したリーン思考を起業に応用することにより、顧客価値にフォーカスして起業のムダな部分を排除しつつ時間とお金を節約し、検証による学びを通して画期的な新製品を開発する手法である。まず挑戦の要となる価値仮説と成長仮説を立案し、図 4.2.8 の構築－計測－学習のフィードバックループを回すことにより科学的に仮説を検証する。「構築する」フェーズにてアジャイル開発により MVP³⁰ を構築し、「計測する」フェーズにて顧客から実測データを集め、「学ぶ」フェーズにて戦略を方向転換（ピボット）するか否かを判断する。このループを素早く回すことにより、イノベーションの成功率を高めるアプローチである。

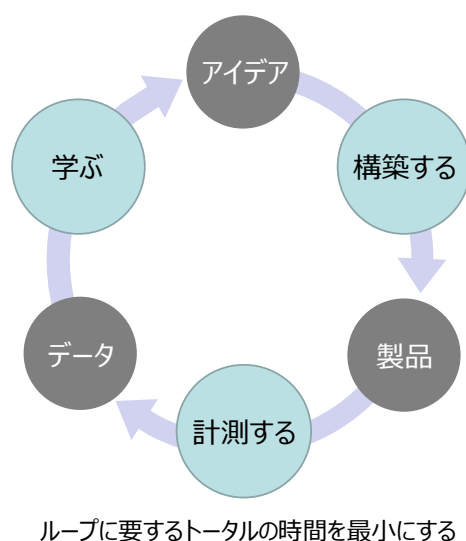


図 4.2.8 リーン・スタートアップの「構築－計測－学習」フィードバックループ
（出典：日経 BP「リーン・スタートアップ」第 2 部の図より⁴⁶）

また最近ではアジャイル開発の原則を組織改革や働き方改革に取り入れる等の動きがあり、プロダクトのデジタル化の流れ（本から電子書籍、物理サーバからクラウド、自動車からコネクテッドカー等）に伴って多くの新製品やサービスがアジャイル開発から生み出されていることから、その存在感を増している。

このようにアジャイル開発とは価値観や原則であり、ソフトウェア開発手法であり、ビジネス手法でもある複雑な概念だが、IPAではその概念構造を「アジャイル開発の家」として整理している³¹。つまり図 4.2.9 に示す通り、土台として組織文化があり、人間中心と技術の尊重の二本柱があり、役に立つ動くソフトウェアを通じてビジネス価値を最大化するという梁と屋根があり、家の中で高速仮説検証サイクル等の活動を行うものである。

⁴⁶ エリック・リース著、井口耕二訳「リーン・スタートアップ～ムダのない起業プロセスでイノベーションを生み出す」、2012 年 4 月、日経 BP

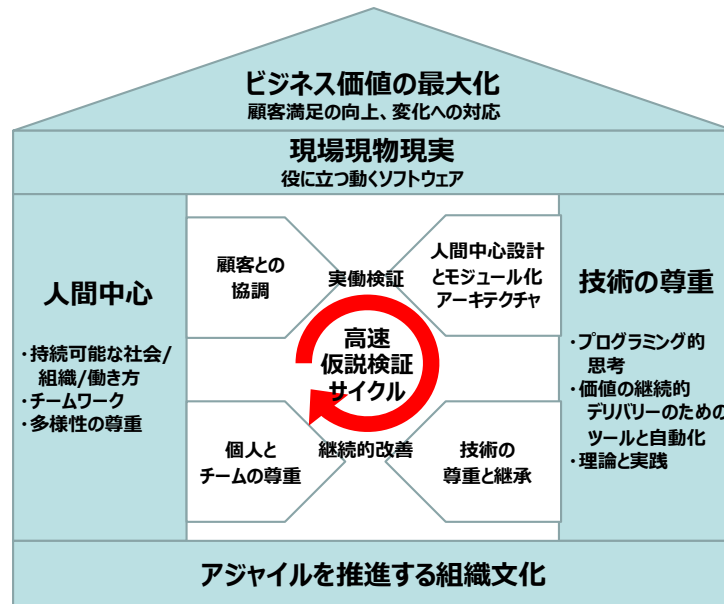


図 4.2.9 アジャイル開発の家（アジャイル開発の全体像）

4.2.3. アジャイル開発により得られる効果

現代社会は、成熟して商品開発が難しくなり、いわゆるディスラプターが他業界から突然参入して市場を席捲することもあり（例：サーバ市場における Amazon）、先行き不透明で将来の予測が困難な時代と言われている。この時代にアジャイル開発が必要とされる理由の1つは不確実性への適応のためである。例えばスタートアップ企業が画期的な新製品アイデアを発明し、図 4.2.10 左図のように納期通りに要件通りのプロダクトをリリースしても顧客を想定通りに獲得できる保証はない。なぜなら人は現物を見るまで自分の欲しい物がよく分からないことが一般的で、実際にそれが提示されてもどのような枠組みで理解、受容するかによっても評価は変わり得る。このような中で、事業を始める段階で先行きを見通すことは不可能だからである。一方、図 4.2.10 右図のように小さく始めて素早くリリースを繰り返し、顧客の反応を確かめながら仕様を柔軟に変更するアジャイル開発の場合、リスクを最小化して成功の可能性を高める効果を期待できる。仮に失敗しても早めの撤退が可能なのでムダな開発コストの発生を抑制できるし、リリース後の継続的な改善により顧客との信頼関係を維持することもできるからである。

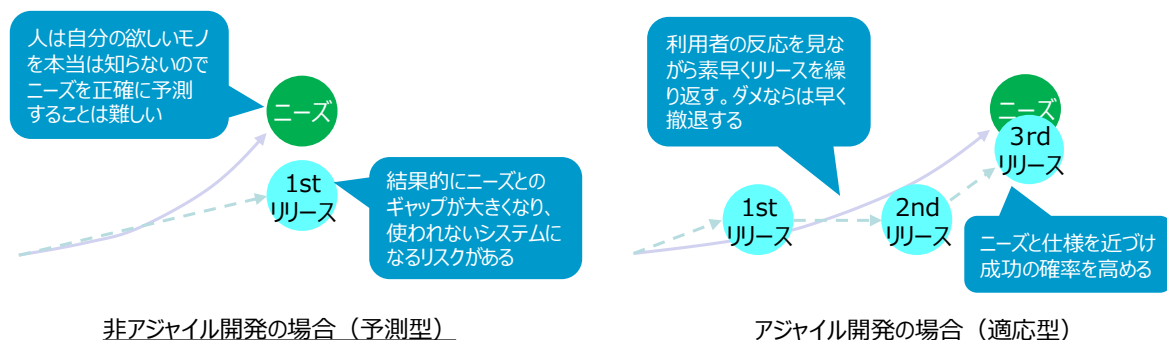


図 4.2.10 非アジャイル開発とアジャイル開発の違い

4.2.4. アジャイル開発の考慮点

図 4.2.9 の「アジャイル開発の家」をブレイクダウンすると、アジャイル開発の考慮点として例えば以下が挙げられる。

大分類	小分類	考慮点
土台	アジャイルを推進する組織文化	・企画、開発、運用の一体化の推進⁴⁷ ・組織へのアジャイルマインドの浸透 ・準委任契約を選択できる調達ルール整備
柱 (人間中心)	持続可能な社会／組織／働き方	・社会貢献、生きがい、働き方改革 ・アジャイル組織、学習する組織、知識創造
	チームワーク	・顧客と対話し、協働して成果を生む ・透明性(かんぱん等)と検査・適応(ふりかえり等)
	多様性の尊重	・様々な専門や経験を持つ多様なチーム、シナジー
柱 (技術尊重)	プログラミング的思考	・論理的推論(演繹と帰納、仮説形成) ・データとアルゴリズム
	価値の継続的デリバリーのためのツールと自動化	・CI/CD、DevOps ・自動化、人の無用な負担の排除
	理論と実践	・スキルの学習、育成、教育 ・技術力をもって生産性と品質、信頼性を担保
家の中	高速仮説検証サイクル	・タイムボックスによる反復型アプローチ ・短いサイクルで顧客が評価できるソフトウェアを提示し、顧客の反応を見て価値を検証
	実働検証	・実際に役に立つ動くソフトウェアによる価値の検証
	継続的改善	・顧客の反応によりプロダクトを継続的に改善 ・状況の変化に応じてプロセスも柔軟に変更
	顧客との協調	・チームと顧客が直接対話しながら、Win Win の関係を構築
	個人とチームの尊重	・自律的なチームで、人の能力を最大限に発揮
	人間中心設計とモジュール化とアーキテクチャ	・良質な顧客体験を提供する UX 設計 ・分業して構築可能なマイクロサービス設計
	技術の尊重と継承	・常に適切な技術とスキルを学習し、それを社会に還元し、次世代に継承する
屋根／梁	現場現物現実	・実際に役に立つ、動くソフトウェアを提供
	ビジネス価値の最大化	・顧客満足度の向上、顧客ロイヤリティの向上 ・競争力の向上、企業価値の増大

このうち最も重要かつ難しいのは「土台」を作る部分である。「アジャイル開発とスクラム第2版」⁴²によると、アジャイル開発の現場で起きている問題として「目的を理解せずに、表面的にスクラムの進め方をなぞるだけというチームが増えている」(出典：アジャイル開発とスクラム第2版 p.68)。例えば、「打合せが形式的で言うべきことを言わない」のであれば、そのチームは確約(お互いのサポートを確約)、公開(課題等の公開)、勇気(正しいことをする勇気)の価

⁴⁷ IPA「なぜ、いまアジャイルが必要か」 p.11、2020年2月、<https://www.ipa.go.jp/files/000073019.pdf>

価値基準を実践していないのでアジャイル開発の家の「人間中心」の柱がない状態であり、もし価値基準の存在を知らないならば土台もないと言えるだろう。同書は他にも以下のような問題例を挙げているが、こちらは確約（ゴール達成等）や尊敬（お互いを尊敬）の価値基準を実践しておらず、同様に土台や柱がない。このように土台や柱がない「家」では、その中での活動に支障を来とし、ビジネス価値の最大化という目的を達成することは難しい。後述する 4.2.5 の先進事例を参考にアジャイル開発の基本的なフレームを踏まえた実践のあり方を検討してもらいたい。

問題	出典元ページ
・スプリントの計画を行う際に、プロダクトオーナーと開発者の間で駆け引きが発生する（プロダクトオーナーが詰め込もうとする。開発者が防御的になる。） ・プロダクトオーナーがプロダクトバックログを作成し、開発者は単にプロダクトオーナーに言われたものを作るだけになっている	69 ページ
・スクラムマスターがプロダクトの成果や組織の目標にコミットメントせず、スクラムのルールを守らせることだけに注力し、スクラム警察化している ・スクラムマスターが雑用係（コーヒーやお菓子を買ってくる）になっている ・スクラムマスターが単に会議の司会役になっている	70 ページ

図 4.2.11 スクラムにて陥りやすい罠（出典：アジャイル開発とスクラム第 2 版⁴²⁾

なお、アジャイル開発の課題として大規模案件への対応が以前から指摘されているが⁴⁸⁾、最近ではアジャイルを大規模化するフレームワークである「LeSS」や「SAFe」の適用事例が国内でも報告されるようになってきた。大規模フレームワークの選択肢も増えており、スクラムの提唱者であるケン・シュエバーが 2015 年に「Nexus」を、もう 1 人の提唱者であるジェフ・サザーランドが 2018 年に「Scrum@Scale」を発表している。2019 年にはプロジェクト管理の世界標準である PMBOK を策定する PMI (米国プロジェクトマネジメント協会) が「Disciplined Agile」として、Scrum や SAFe 等の複数のフレームワークを組み合わせ適用する考え方を示している。

4. 2. 5. アジャイル開発の先進取組み事例

アジャイル開発の先進的な取組み事例を 4 つ紹介する。A 社は前出の製造業の事例、B 社はパートナーメインから内製化にシフトした事例、C 社はリーン・スタートアップの事例、D 社は IT ベンダー側から見た事例である。各社共通しているのは「アジャイル開発の家」における土台と柱を具現化している点である。つまり、現状と常識に捉われず日々改善と成長を志すマインドセットを持ち（土台）、経営者、プロダクトオーナー、開発者等が会社や立場の違いを超えて協力することにより相乗効果を発揮している（人間中心の柱）。さらにチームとしてビジネスと IT 両方のスキルを備えることにより（技術尊重の柱）、ビジネス価値の最大化という屋根と、現場現物現実という梁をかけることに成功している。

⁴⁸⁾ 経済産業省「デジタルトランスフォーメーションに向けた研究会（第 3 回）議事要旨、2018 年 7 月 5 日、https://www.meti.go.jp/shingikai/mono_info_service/digital_transformation/pdf/003_gijiyoshi.pdf」によると「アジャイルのイメージとして、比較的小規模で世の中の動きに対して迅速・柔軟に対応できるものに適していると思われるが、将来的に大規模なものに適用するにはどうすればよいかを検討して経験を積み重ねていく必要があり（後略）」とある

（事例１）製造業Ａ社

基盤のプレス加工を得意とする同社は製造現場の作業日報のペーパーレス化を検討していたが、以前作ったシステムが現場に定着しなかった苦い経験があるため、従来帳票をタブレットに置き換えるだけのＩＴベンダー提案に納得せず、時間をかけパートナーを探してアジャイル開発と出会った。社長が現場の使い易さ（UX）に熱い思いを持っており、現場を巻き込みアイデアを出し合い、スマホでＱＲコードを撮影してストレスなく作業を記録するシステムを開発した。以前から自社販売に進出したいと考えていたが、パートナー会社と双方の専門分野の強みを活かし、シナジーを発揮して事業や活動を拡大している。

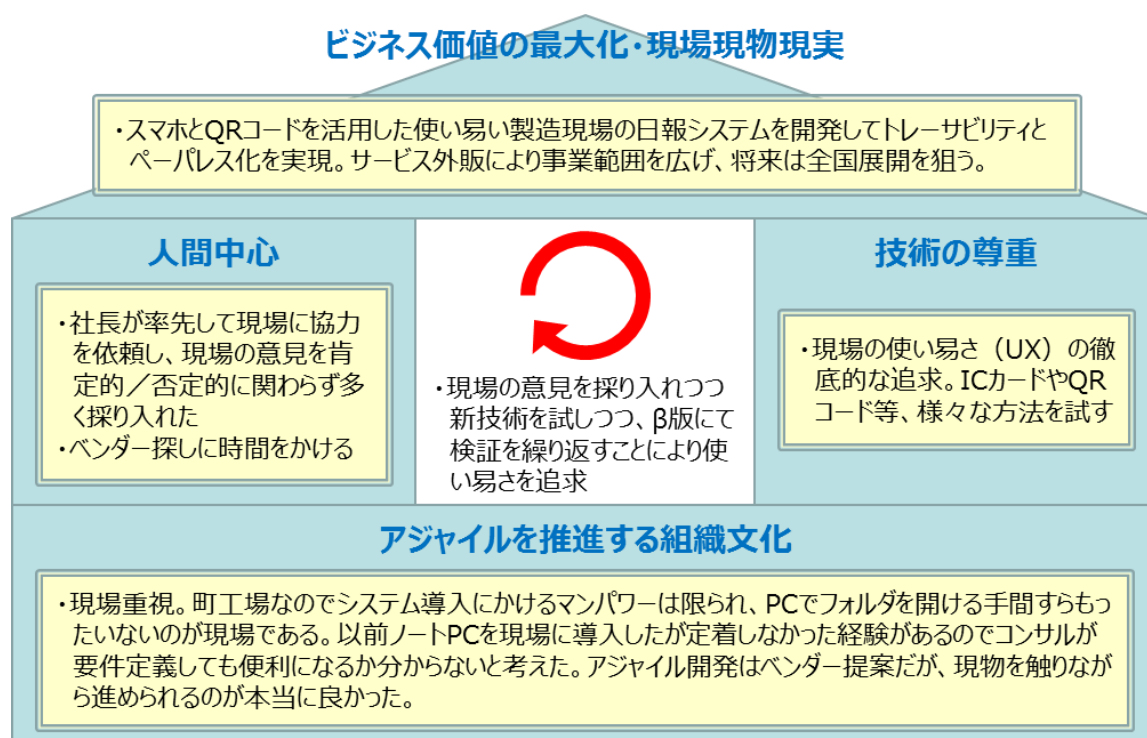


図 4.2.12 製造業Ａ社のアジャイルの家

（事例２）宿泊業・飲食サービス業Ｂ社

リゾート・温泉旅館の経営及び運営受託を行う同社は順調に業績を拡大しており、システム部門に対する経営陣・事業側からの要請や期待も高まっていた。しかし、従来のパートナーをメインとする体制では小さな機能追加の手続きに数か月かかるため期待に迅速に応えることが難しく、内製化＋アジャイル開発にシフトした。最初は１人のみ採用して、経営陣・事業側の期待に迅速に応えることによりアジャイル開発の良さを理解してもらいながら、徐々に体制を拡充した。チームは経営陣や現場と一緒に価値を生み出すという共通の目的の下で年齢や社会人歴等が異なる現場出身者と中途入社エンジニアが自律的に活動しており、毎月の会議では担当者が経営陣と直接会話する等経営陣との距離も近い。チーム編成はアジャイル開発手法の１つであるスクラムをベースにしているが、チームによっては教科書通りのスクラムを行うのではなく、スクラムを参考にしながらも、チームが自己組織化するように工夫している。スキル面ではモデ

リングや UX 等の社員教育に力を入れており、チームのスキルマップを作り、メンバーは主体的にスキルを習得している。直近数年は新型コロナ禍の中で三密回避の仕組み構築や GOTO トラベルキャンペーンの仕様変更対応、ギフト券のふるさと納税対応等のイベントが連続したが、経営陣や顧客の要請に迅速に応えることが出来た。

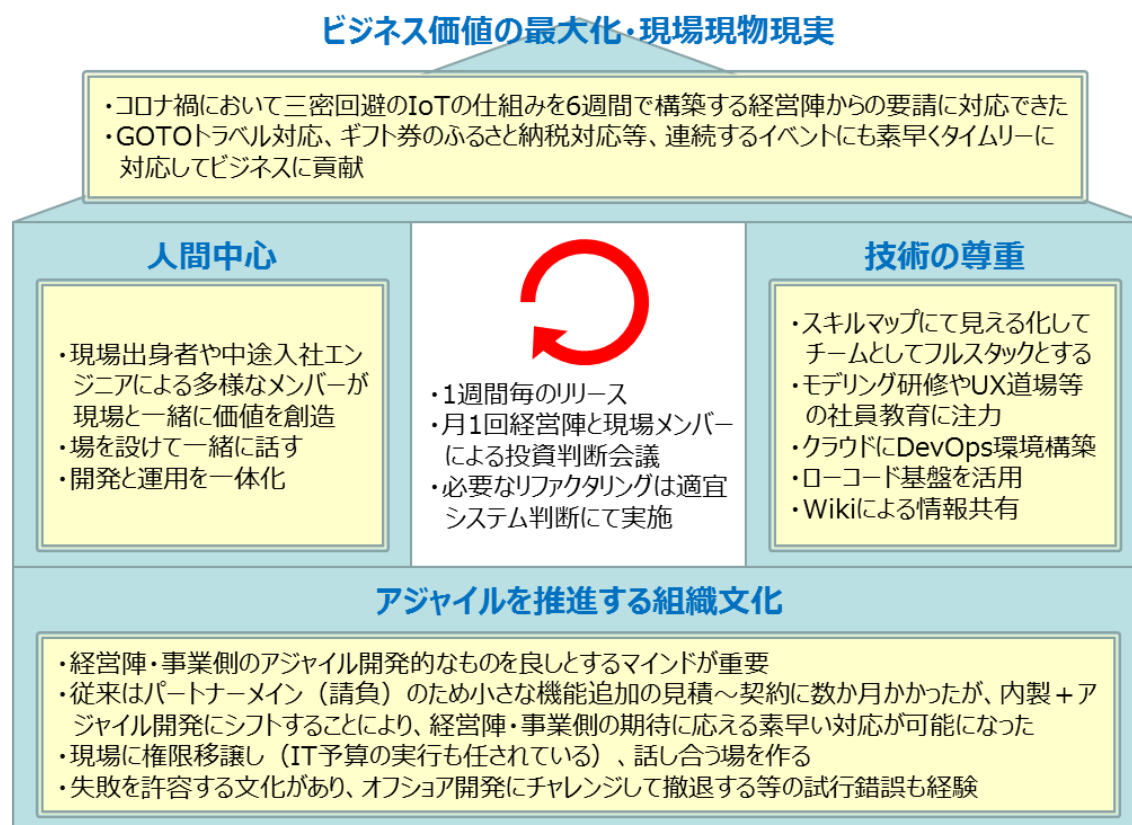


図 4.2.13 宿泊業・飲食サービス業B社のアジャイルの家

（事例3）職業紹介サービス業C社

同社は IT／WEB エンジニア向けの転職・就職・学習サイトを開発・運営しており、個人がキャリアを意思決定する時代の流れの中で新しいビジネスモデルを考案してリーン・スタートアップにて検証を繰り返し、55 万人が利用する国内最大級のプラットフォームとして成長した。市場の課題やビジネスモデルを社員全員が理解して定量的に検証する文化形成を目指しており、そのため毎月の会議では経営者が目指す姿やステップ毎の世界観を示し、経営数値等の情報もオープンにしている。KPI 自動集計システムやデータ分析環境を整備しており、エンジニアでなくても SQL を使って定量的に分析できる。またシステム部門がコストセンターからプロフィットセンターに移行する時代の流れの中でエンジニアにも事業開発やビジネスのスキルが必要と考えている。例えば事前に AB テスト⁴⁹にて検証してムダな機能を「作らない」発想があれば、無駄なコスト発生を抑制できるからである。またコンピュータサイエンスの基礎力も重視する。なぜなら基礎力の高い人はコンピュータの振る舞いを理解しているので不具合が発生しても素

⁴⁹ AB テストは顧客体験の調査手法であり、例えば WEB 広告にて顧客に A 案／B 案をランダムに表示し、その有効性を統計的仮説検定等の手法を用いて判断する。

早く対策可能であり、自学自習能力も高いので新技術習得も速いからである。このような価値仮説がエンジニアや採用企業に広く受け入れられ、エンジニアの基礎的スキルをコーディングテストにて見える化するサービスに繋がっている。

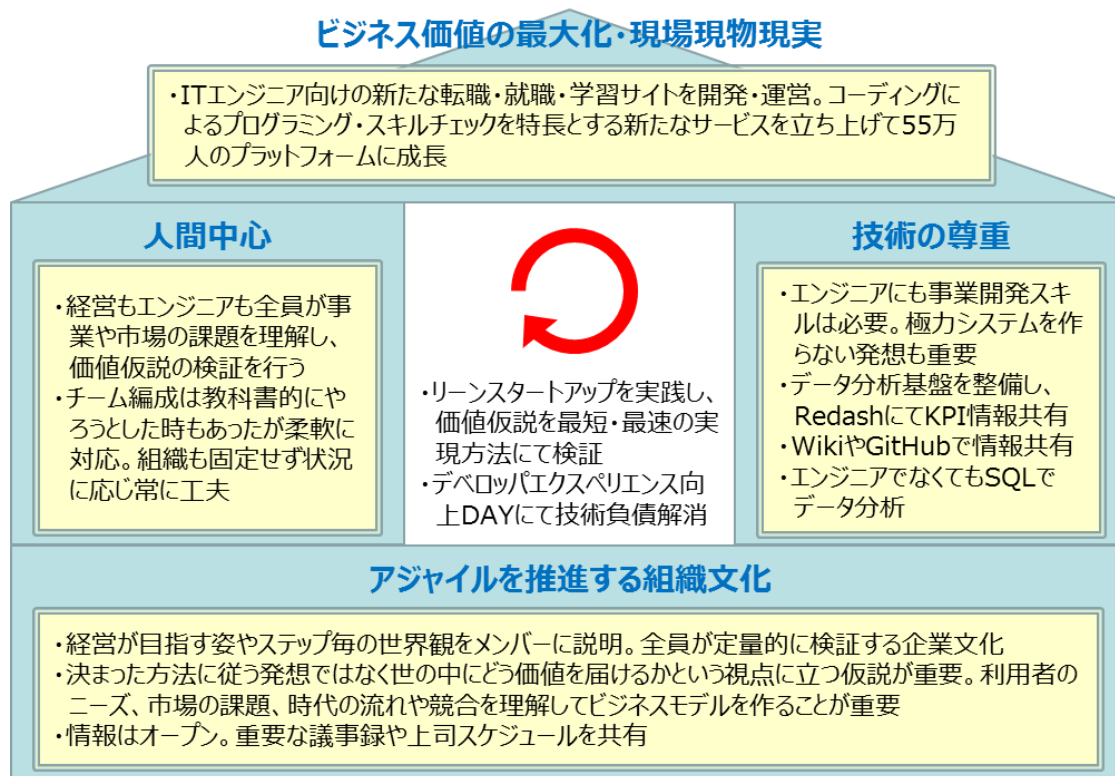
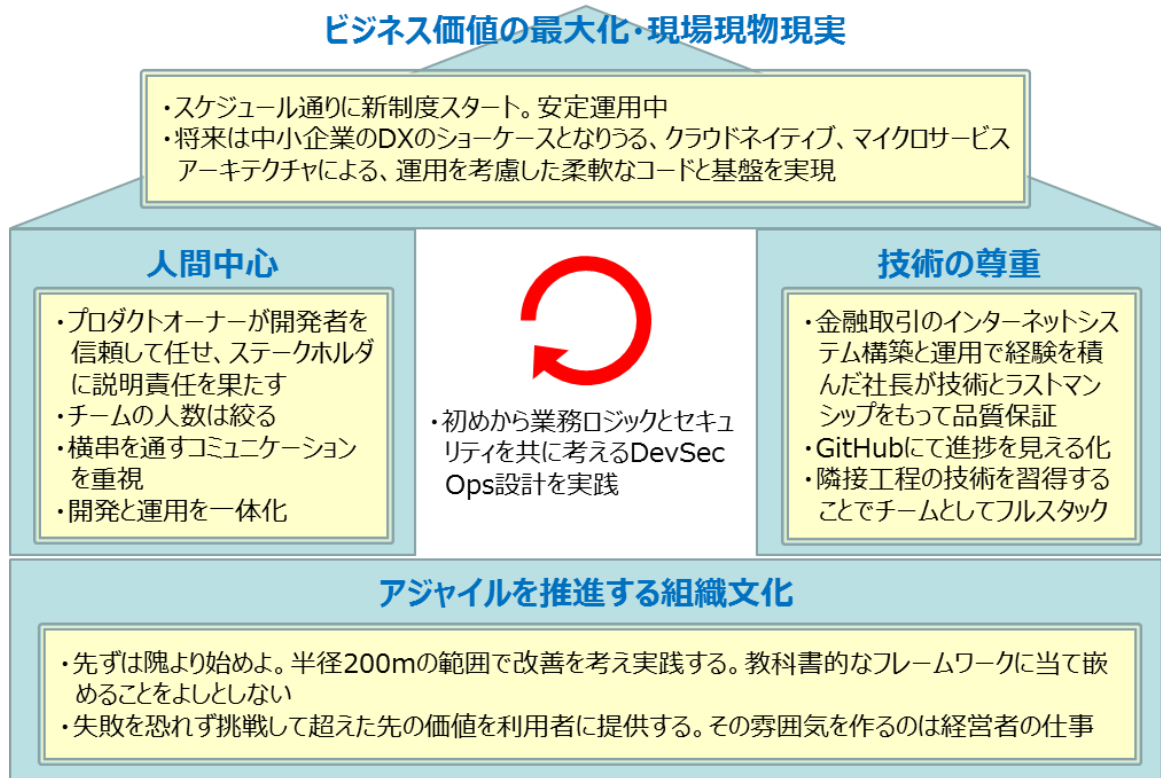


図 4.2.14 職業紹介サービス業C社のアジャイルの家

(事例4) 情報サービス業D社

同社は新制度の開始に伴うインターネット経由にて申請受付等を行う案件において、納期の変更ができないアジャイル開発という難しい制約のある中で先進的なクラウドとマイクロサービスによる柔軟なコードと基盤の構築にチャレンジし、予定通りにリリースして安定的に運用している。当該事例では発注側と受注側が相互にリスペクトして信頼関係を築いたことが成功要因として挙げられる。プロダクトオーナーはステークホルダに説明責任を果たし、技術面は開発者を信頼して任せる。一方、開発者は金融取引のインターネットシステムの経験にて培った知見を活かして将来的な運用を視野に入れて開発を行い、開発状況の透明性を確保して検査・適応のサイクルを上手く回すことによりプロダクトオーナーの期待に応えている。またスキル面では各メンバーが最低 2 つの隣接技術を習得することによりチームとしてフルスタックになることを目指している。隣接技術とはバリューチェーンの隣り合う工程、例えば開発者ならばデザイナー界隈の用語・知識・都合を解釈し、自分なりに言い換えてコミュニケーション出来る能力の意味であり、そのため読書等の基礎的な学びの力が重要と、D社は考えている。



4.3. スピード・アジリティを支えるマイクロサービスアーキテクチャの

活用

本項では、あるべき IT システムの中の**競争領域の独自アプリケーション(A)**の構築に関する技術として、マイクロサービスアーキテクチャについて、活用することで実現できる効果や考慮点などについて、説明を行う。

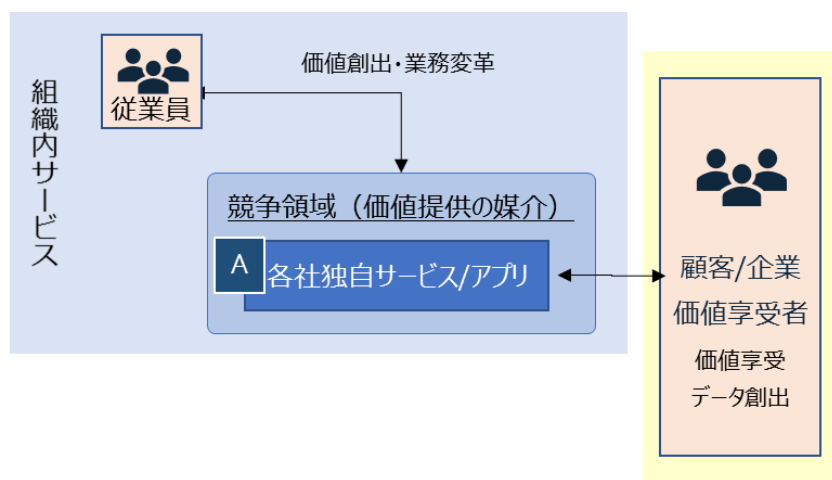


図 4.3.1 あるべき IT システムにおける**競争領域の独自アプリケーション(A)**

4.3.1. マイクロサービスアーキテクチャを採り上げる背景

4.1 DX を実現するための技術要素群の全体像 でも述べたように、**競争領域の独自アプリケーション(A)**は、変化に対応し、スピード・アジリティを備えた IT システムである必要がある。独自アプリケーションの構築に関して考慮すべき要素としては、どのような言語を選ぶか、どのようなツールを使うかなど、様々なものがあるが、その中で、アプリケーションの構造を決めるのは、アーキテクチャである。いかなるアーキテクチャを採るかによって、修正の容易さや影響範囲などが変わり、工数や期間へ影響が出ることとなるため、要求仕様の変化に柔軟かつ俊敏に対応するためには、アーキテクチャの決定が大きな要素となる。

そうした視点から、スピード・アジリティを実現するために適したアーキテクチャとして、マイクロサービスアーキテクチャを挙げることができる。マイクロサービスアーキテクチャとは、1つのアプリケーションを構成する各機能をサービスとして分割し、各サービスが1つのアプリケーションとして動作するようにしながら、それが協調的に作動することで（こうした構成を疎結合という）全体としてやはり1つのアプリケーションとして挙動するように構成するアー

キテクチャ⁵⁰を言う。それぞれのサービスが機能として独立するだけでなく、データについてもサービス毎に分割して管理することが望ましいとされている。⁵¹

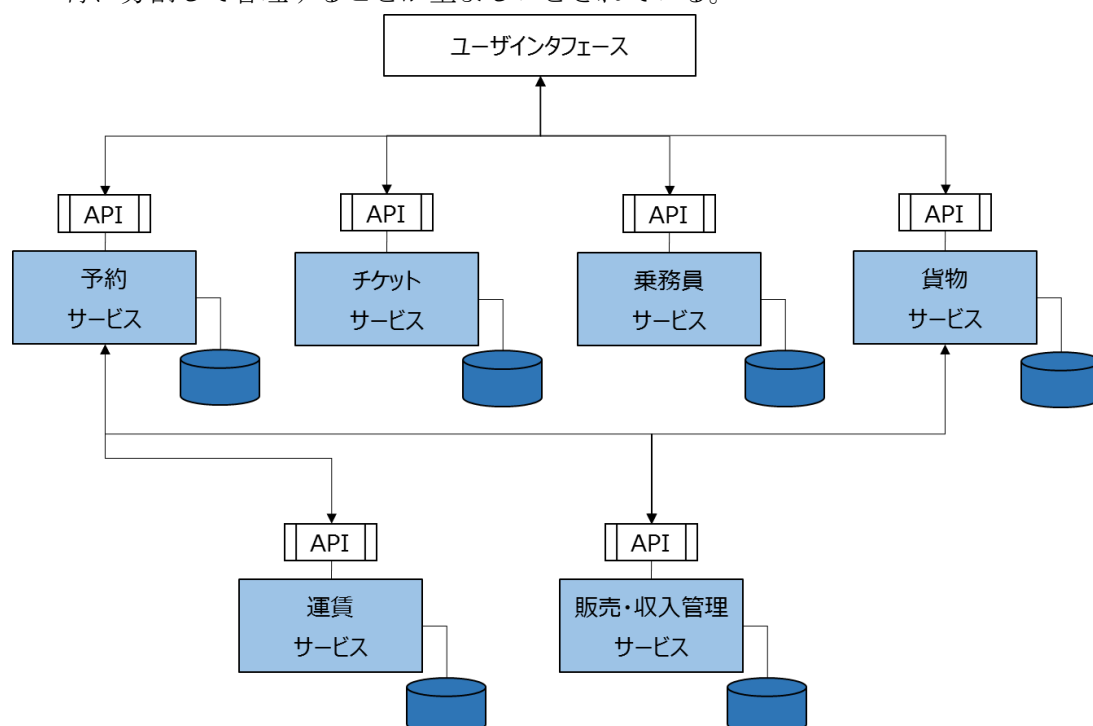


図 4.3.2 マイクロサービスの構成例

そうしたマイクロサービスアーキテクチャを IT システムにおけるスピード・アジリティの実現に関連する技術として、採り上げる背景としては、(1)モノリス構造の限界、(2)マイクロサービスアーキテクチャを支える技術の整備、(3)開発方法論の確立の3点を挙げることができる。

(1)モノリス構造の限界

1点目は、IT システムの複雑化、およびユーザーの要求や市場の変化のスピードが速くなってきたことで、従来のモノリス構造のアプリケーションでは、ビジネス/環境の変化に応える上での課題が、見過ごせないものとなってきたという点である。例えば、従来型のモジュール化技術によるアプリケーションでは、修正する際に、影響する範囲が複数の範囲に渡って、影響調査/修正/テストにかかる工数が増大し、修正内容に比してコストや期間が必要とされるようになってきた。また、修正後にアプリケーションをリリースする際に、一部の機能の修正に対しても、他の機能を含めたアプリケーション全体のリリースが必要であるため、リリースに対する作業負荷が増加することとなった。

⁵⁰ Martin Fowler, "Microservices a definition of this new architectural term", <https://martinfowler.com/articles/microservices.html>

⁵¹ マイクロサービスアーキテクチャに関する詳細は、以下の資料も参照されたい。

Sam Newman(著)、佐藤直生(監訳)、木下哲也(訳)『マイクロサービスアーキテクチャ』オライリー・ジャパン、2016年

Chris Richardson(著)、長尾高広(訳)、樽澤広亨(監修)『マイクロサービスパターン』インプレス、2020年

(2)マイクロサービスアーキテクチャを支える技術の整備

2 点目としては、以下に挙げる技術が整備されてきたことで、システム開発の際にマイクロサービスアーキテクチャをより活用しやすくなってきたことがある。

クラウド基盤の普及：マイクロサービスアーキテクチャを使った開発に必要なインフラ環境を短期間で準備することが可能。また、構築した IT システムの運用時にも、処理量の変化に対応してマイクロサービス毎にスケールアウト/インを柔軟に行うことができる。運用に関しては、それ以外にも、クラウド上で稼働するアプリケーションを補助するための機能が提供されている。

コンテナ環境の整備と機能の充実：アプリケーションの実行環境の仮想化、それを管理する機能により、異なる環境間へのアプリケーションの移行がより容易になった。それにより、マイクロサービスにより構成されたアプリケーションの開発環境からテスト環境への移行や、テスト環境から本番環境へのリリースを、より効率的に進めることができるようになった。

CI/CD⁵²(継続的インテグレーション/継続的デリバリー)を前提とした開発/テスト、リリース：アプリケーション修正時のテスト自動化、修正したアプリケーションの本番環境への反映の省力化により、マイクロサービス毎に品質の確保を行いながら、短いサイクルで開発/テスト、テスト環境から本番環境へのリリースを、行うことができるようになった。

Web API：各マイクロサービスが連携して、動作することを実現するためのインタフェースとなる。汎用性の高いプロトコルとなっており、異なるプログラミング言語で作られたアプリケーション間においても、機能的に連携することができる。そのため、既存のアプリケーションなどの資産の活用や、外部が提供するサービスの利用なども可能となる。

サービスメッシュ：各マイクロサービスからサービス間で通信する部分を肩代わりし、サービス間の通信の管理/制御を行う。それにより、サービス間の通信の可視化、トラフィックの制御などを行う。

(3)開発方法論の確立

3 点目は、アプリケーションの開発手法として、マイクロサービスアーキテクチャと親和性の高いアジャイル開発の手法が使われるケースも増えてきたという点となる。開発プロジェクトの進め方としても、要件に不確定な要素を含みながら、ビジネス/環境の変化に対する柔軟な対応を行うため、そうした開発の仕方が使われるようになってきた。

以上のように、ビジネス面からマイクロサービスアーキテクチャを活用する必要性が増してきたことと共に、技術面でもマイクロサービスアーキテクチャを採用した IT システムの構築、運用を行うことが、実用的なものとなってきたことが、当技術を適切なアーキテクチャとして採用した理由である。

⁵² Continuous Integration／Continuous Delivery の略

4.3.2. マイクロサービスアーキテクチャの活用により実現できる内容

マイクロサービスアーキテクチャを活用し IT システムを構築することにより、実現できる内容について、説明する。

①サービスを単位として各機能が分離・独立していることによる俊敏な開発の実現

マイクロサービスアーキテクチャを採用して、アプリケーションを設計/開発することで、複数のサービスを、それぞれ分離・独立し、互いに疎な関係とすることができる。そのため、俊敏な開発やサービスごとに分割した構成での IT システムの構築を行うことができる。こうした開発では以下が実現可能となる。

- ・各サービスの独立性が高いため、一部の機能に修正が発生した場合でも、他のサービスに対する修正を発生させず、影響範囲が極小化され、修正に必要な作業量が抑えられる。
- ・修正後のテストについても、修正したサービスをメインとして、テスト範囲が限定され、テストに要する期間やコストも少なく済む。
- ・修正したサービスを中心に、テスト対象を絞ることができるため、品質の低下を抑止できる。

②Web API を介したサービス間の連携による柔軟性を備えた構成の実現

疎結合状態の各サービスは、Web API として REST、gRPC などの比較的軽量なプロトコルを利用し、1つのアプリケーションとして連携して動作する構成をとり、以下を実現できる。

- ・サービスの入替を伴うような構成の変更に対しても、インタフェースを合わせることで、大きな影響を発生させず、柔軟な対応が可能となる。

4.3.3. マイクロサービスアーキテクチャを活用する上での考慮点

マイクロサービスアーキテクチャを用いて、IT システムを構築し運用していく中で様々な課題が発生している場合がある。実際に IT システムを運用・管理していく観点から設計・構築上、考慮が必要な点が存在する。

課題が発生する主な理由としては、マイクロサービスアーキテクチャの特徴によるところが大きい。それは、1つは多数のサービスに分割/分散されていることにより、構成自体や連携する動作が複雑化することによる。もう1つは、それぞれサービスとして独立性が高いがために、それぞれのサービス内での対応が見過ごしされがちな点があることによる。

前者の例としては、トランザクションの管理が挙げられる。

これまではトランザクションの管理などは、トランザクションを制御するミドルウェアおよび RDB(Relational Database)などの組み合わせ、さらにはそれを利用したアプリケーション側で一意的に担保されてきた経緯がある。しかしながら、マイクロサービスアーキテクチャを意識し、

その特性に配慮したうえで活用できるトランザクションサービスがまだ成熟しきっていないこと、機能分散が始まると一つのトランザクションとして制御を行うことが予想外に難しいことなどがあり、留意をして設計を進める必要が出てくる。

後者の例としては、証跡を残すためのログの出力など、非機能要件に関するものが挙げられる。

モノリシックなアプリケーションでは、全体で共通で対応していたログの出力などの対応を、マイクロサービスにおいては、それぞれのサービスが独立しているため、各サービスの中で対応しなければならない。各サービスで、そうした非機能要件に関する対応が、漏れることがないように、統制を利かせるための対策が必要となる。

以下に、マイクロサービスアーキテクチャを採用して IT システムを構築する上で、あらかじめ検討しておくべき点について、説明する。

(1)計画の段階で明示的に検討しておくべき事項

<適用対象>

IT システムの構築にあたり、マイクロサービスアーキテクチャを適用する対象としては、継続的に変更が発生し、スピードが求められるものが、適している。初期構築の際には、従来構造に比してコストのかかる部分があるが、変更時にはコストが抑えられ、マイクロサービスアーキテクチャ活用のメリットを生かすことができる。

また、IT システムの規模についても、小規模ではないものが、適している。それぞれのサービスが独立しており、サービス毎に最低限実装が必要な機能がある。そのため、IT システム全体として、一定程度の規模があることで、マイクロサービスアーキテクチャを使った効果を得ることができる。

<開発体制>

体制としては、1つのチームが、1つのサービスを開発し、その運用を行うことで、マイクロサービスアーキテクチャの活用について有効性をより引き出すことができる。

各サービスは独立しているため、開発/運用する体制としても独立し、開発/運用を通して担当することで、習熟度が上がり、生産性の向上も期待できる。

各チームが独立しているということは、その分、自由度も増すことになるため、監視など IT システム全体としての統一が必要な部分は、組織として横断的にチェックする機能も必要となる。各チームに必要なガバナンスを利かせながら、全体を運営していくことが重要となる。

(2)設計の段階で明示的に検討しておくべき事項

<データの配置>

各サービスとデータベースの配置の仕方としては、大きく 2 種類の方法が取られる。

データベースをサービス毎に保持する方法(Database per Service)と、1つのデータベースを各サービスで共有する方法(Shared database)がそれにあたる。

前者のサービス毎にデータベースを配置する方法は、サービスの独立性を高める上でも推奨されるやり方となるが、複数のデータベース間での厳密な整合性が必要とされる場合は、

適用が難しい。業務的に、結果整合性⁵³が許容される場合は、当方法を採用できる。

後者のデータベースを共有する方法は、各サービスが行ったデータ更新について、同期が必須とされる業務において、有効な手段となる。ただし、データベースを共有することで、あるサービスにおいて、データベースに関わる部分に変更が発生する場合、他のサービスへの影響有無について、確認が必要となる。

上記のように、業務的な要件を見極めながら、データの配置については検討する必要がある。

<データの整合性>

サービス毎にデータを管理している場合、トランザクションレベルで整合性が必要なサービスは1つのサービスにまとめるか、複数のサービスにまたがる場合は、それらの整合性を管理するサービスを用意する必要がある。

トランザクションとして整合性が必要な処理としては、サービス毎にそれぞれが管理するデータの更新処理を行っていく。ただし、その中の一部のサービスの更新処理が失敗した場合、不整合な状態となる。そのため、該当のトランザクションにおける不整合を回避するため、トランザクション全体の状態を監視し、一部の処理が失敗した時には、該当のトランザクションに含まれる別サービスの処理を取り消す機能の実装が必要となる。

<サービスの単位>

それぞれのITシステムにおける、適切なサービスの粒度を最初から見極めることは難しく、初期構築の段階では、大きく始める方がリスクは低いとされている。その上で、運用を進めていく中で、変更が頻繁に発生する部分を、別のサービスとして分割をしていき、適切な粒度へ分解していく。

なお、モノリスファーストとして、最初はモノリス構造で全体を構築し、段階的にサービスを切り出していくという手法も、提案されており⁵⁴、最初から全体をマイクロサービスで構成するのではなく、ステップを踏んで取り入れていくという方法も、取り組み方の1つとして考えられる。

(3) 開発・テストの段階で明示的に検討しておくべき事項

<プロセス>

俊敏性を満たした開発を進めていくためには、以下の要素をプラクティスとして取り入れることも、必要となる。

アジャイル開発：機能の改修が多く、繰り返し行なわれるようなITシステムについては、アジャイル開発の手法を取ることで、短期間でのリリースの実現に寄与することになる。アプリケーションの構造がサービス毎に分割されており、体制としてもサービス毎に組成し、チーム毎にアジャイル開発を進めることで、他のサービスの影響を受けずに、並行して開発を進められ、ITシステム全体としても、効率的な開発を実現できる。

⁵³ ある時点では複数のデータ間で不整合な状態となっており、最終的に整合性が取れた状態となっていること

⁵⁴ Martin Fowler, "MonolithFirst", <https://martinfowler.com/bliki/MonolithFirst.html>

CI/CD：開発したソースについて、コンパイル、ビルド、テストから、リリースする一連の流れを自動化し、短い期間での対応、品質確保へつなげる。

<成果物>

マイクロサービスアーキテクチャを使った開発を行う場合、他のサービスと情報を共有しなければならない部分が、ドキュメントとして作成すべきものとして、重要となる。

マイクロサービスアーキテクチャでは、サービス間で連携して、動作することが基本となっているので、それらの間でのインタフェースとなる API 仕様を表す設計書は、必須となる。また、データベースを共有している場合は、データベースの定義書や関連図も必要となる。

<テスト>

単体テストからシステムテストに至るテストに関して、従来型の IT システムにおける考え方と大きく変わるわけではない。

サービスの修正を行った場合、サービス内の個々の機能については、単体テストとして<プロセス>の項目で記載した CI 環境の中で、自動化しホワイトボックステストにより、確認を行う。

また、サービス間についても、正常系のテストなど可能な部分は、自動化により、機能連携の確認を行う。自動化によりカバーできない部分は、実施範囲、確認内容を決めて、テストを行う。

(4)非機能に関して明示的に検討しておくべき事項

<性能対策>

マイクロサービスによる構成は、分散システムとなるため、適切にサービスを管理するため、サービスメッシュの機能を活用することも、重要な要素となる。

サービスメッシュにより、サービス間のトラフィックコントロールや、サービスディスカバリ⁵⁵と負荷分散による、制御を行うことができる。

また、一部のサービスに障害が発生した際に、対象のサービスへの無駄なトラフィックを発生させないように、サーキットブレーカーにより障害部分を切り離す機能も備えている。

複数のサービス間での性能の問題が発生した際の切り分けとして、テレメトリデータの収集機能も活用することができる。

<性能拡張性>

非機能面では留意したい要素の一つに性能拡張性がある。マイクロサービスにより構成された IT システムではアクセス要求に対してリニアに拡張ができるとされ、それがメリットであるとされているが、それは分散処理が可能な演算を中心とした処理の話が中心である。ひとたびデータアクセスが介在すると、データの属性によってはデータベースへのアクセス中に排他（ロック）がかかるケースもある。更新系が多く実行される場合、排他の範囲が大

⁵⁵ 呼び出すサービスの NW 上の位置を特定すること

きくなれば参照系にも影響が出るため、マイクロサービスでも同様な配慮が必要となる。必ずしも最新データである必要が無い参照系はデータベースを分ける、アプリケーション側でアクセス優先制御を実施するなど、これまで用いてきた問題解決の方法も有効なものとなる。

<セキュリティ>

特に外部が提供するマイクロサービスを呼び出して利用する場合や、逆に外部から自らが提供するマイクロサービスを呼び出して利用してもらう場合など、連携しているすべてのマイクロサービスを独自に管理できないケースでは、セキュリティについて留意が必要である。

例えば、呼び出す先のマイクロサービスが詐称されていたり、自身が提供するマイクロサービスが意図しない相手に不正に利用されたりすることも考えられる。ネットワークアクセスの制限や認証認可の仕組みを利用して、接続する双方の真正性の確保を行うことが必要となる。

ネットワークで制限する場合は、呼び出しをする側と呼び出しをされる側を同一の閉域網内に実装するといった対処を行う。もしくは、そうした実装が難しい場合には、認証認可の仕組みを利用し、的確な証明書と鍵の管理下で、呼び出しをする側・呼び出しをされる側とも互いに真正性を確認しながら運用をするといった対処を行う。

4.1.4(3)で述べた設計時点におけるセキュリティ機能の作り込みの重要性は、マイクロサービスの実装においても同様である。脆弱性の混入を防ぐコードの共通化、乗っ取られたマイクロサービスの被害波及を防ぐアクセス権限の最小化、コードレビューの強化、等が有効と考えられる。

<信頼性>

信頼性に関する事項としては、以下に留意する必要がある。

外部と連携しながら、1つの機能を実現する場合、障害などで呼び出している外部のサービス側が停止した場合等に備えておくことが必要となる。外部のサービスの呼び出しに対する、応答がない場合のタイムアウト、リトライなどのアプリケーションの動作や停止に関する情報の取得について、あらかじめ設計に含めておく必要がある。

また、サービスが地域をまたがって稼働しているケースで、一部のサービスが稼働している地域に大規模災害が発生し、該当地域のサービスがすべて停止することも起こりうる。特にミッションクリティカルなITシステムでは、そうした有事の際のサービスの切り替え、復旧に対する設計、準備、訓練を行っておくことも、信頼性を確保する上で重要となる。

実際には、上記のような事前に予想できないサービス停止の他に、計画的なメンテナンスタイミングの確保もITシステムの安定的な稼働の継続には、欠かせないものとなるため、計画停止を可能にする構成、メンテナンスの仕組みの構築が必要となる。

<監査・監視>

監査・監視、システムログの取得が、運用に関わる内容として、考慮が必要となる。

機能が各サービスに分散されているがために、1機能の処理時にも、各サービスが連携し、複雑な動作が行われる場合も少なくない。1つのサービスから、複数のサービスが呼び出さ

れ、呼び出されたサービスがさらに他のサービスを呼び出すなどサービスの連鎖が行われることもありうる。平時からサービスや機能全体の構造を把握し管理しておき、そうしたサービスにおいて障害が発生した場合、処理の流れをトレースできる仕組みを整えておくことが、原因調査やその後の復旧・修正対応に必須となる。

そのためにも、IT システム全体で証跡を残すことに対する基準やガイドラインを整備し、各サービスで残すログの出力内容、タイミングなどを標準化しておくことが、運用品質の確保にもつながることとなる。例えば、後にセキュリティ上の要求から追跡調査が必要な場合や監査証跡が求められるような IT システムではどのようなログをどのような形でどこに残すのか、取得タイミングや保存期間、外部媒体への保存などの観点で整理をしておくことが品質の向上につながりやすい。かつ、そうしたログの取得を含め、全体での監視・監査の仕組みの確立が、重要な要素となる。

4.3.4. マイクロサービスアーキテクチャの活用事例

マイクロサービスアーキテクチャや関連する技術、開発手法などを活用して、IT システムを構築した事例について、以下で紹介する。

(1)運輸業 A 社

<背景と IT システムの概要>

顧客の情報をシームレスにつないで、効率的に活用することで、顧客一人一人に対応したサービスの提供を可能にし、顧客体験価値の向上を実現することが、経営上の理念として掲げられていた。そのため、既存業務で管理している顧客の情報を統合し、統合したデータを横断的に、またリアルタイムで、顧客や社内のスタッフが利用できるデータ活用基盤の新規構築を行った。

アプリケーションは、マイクロサービスアーキテクチャを採用して、機能毎に分割し、それぞれを API でつなぐ構成とした。社内の係員が使用する端末や顧客が利用するスマートフォンなどの各チャネルとの接続、ビジネスロジック、統合データベースと連携して、データを共有して活用できる環境を構築した。

<得られた効果>

①俊敏な機能修正

マイクロサービスによりアプリケーションを構築していたことから、新規構築後の保守開発では、修正範囲を限定するとともに、アジャイル開発を適用して、修正の俊敏化につながった。

②構成変更への対応

データ活用基盤の構築後も、顧客との接点の拡大、新たなサービスの提供のため、チャネルやビジネスロジックにおける追加が発生した。その際も、それぞれを担うマイクロサービスが疎結合な構成をとっていることから、所要のマイクロサービスの修正（機

能等の追加) に関しても他への影響を最小限に留め、スムーズな対応を行うことができた。

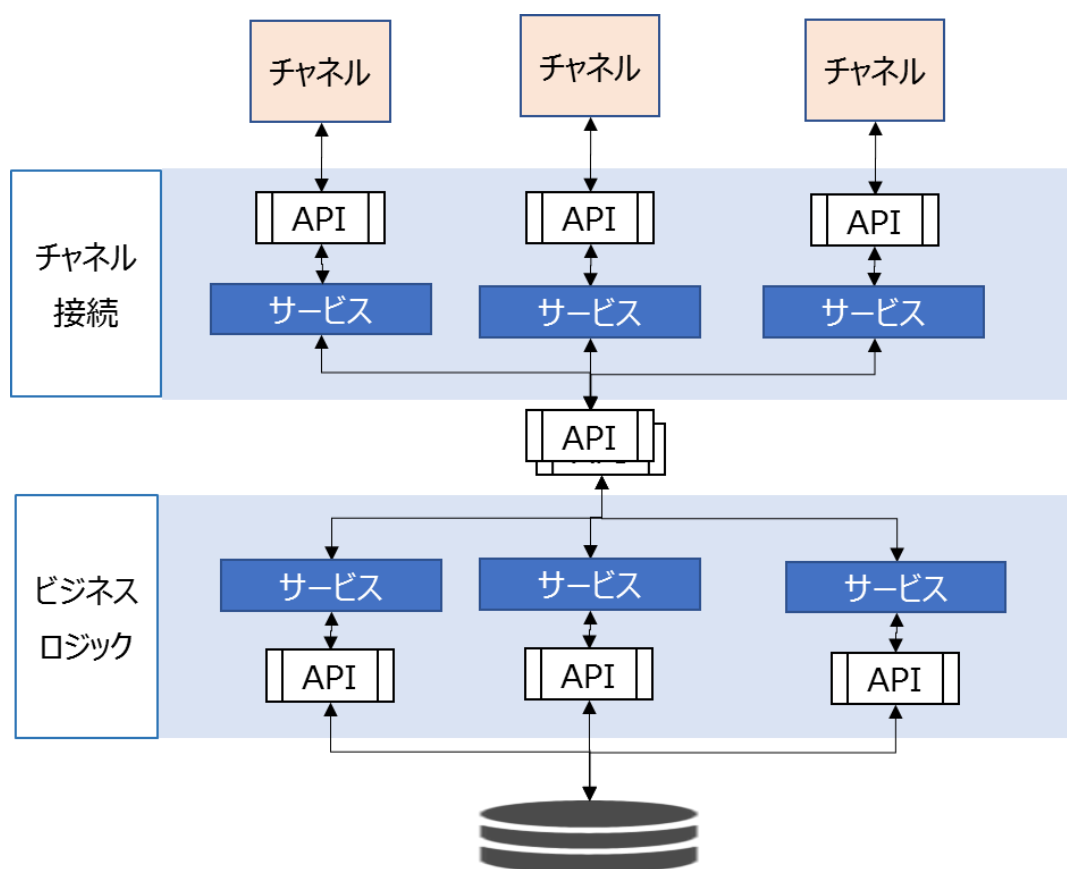


図 4.3.3 A 社のデータ活用基盤概要図

(2) 宿泊業・飲食サービス業 B 社

<背景と IT システムの概要>

顧客へ新たなサービスを提供するため、宿泊予約に関わる機能の新規開発を 9 か月の期間で行った。

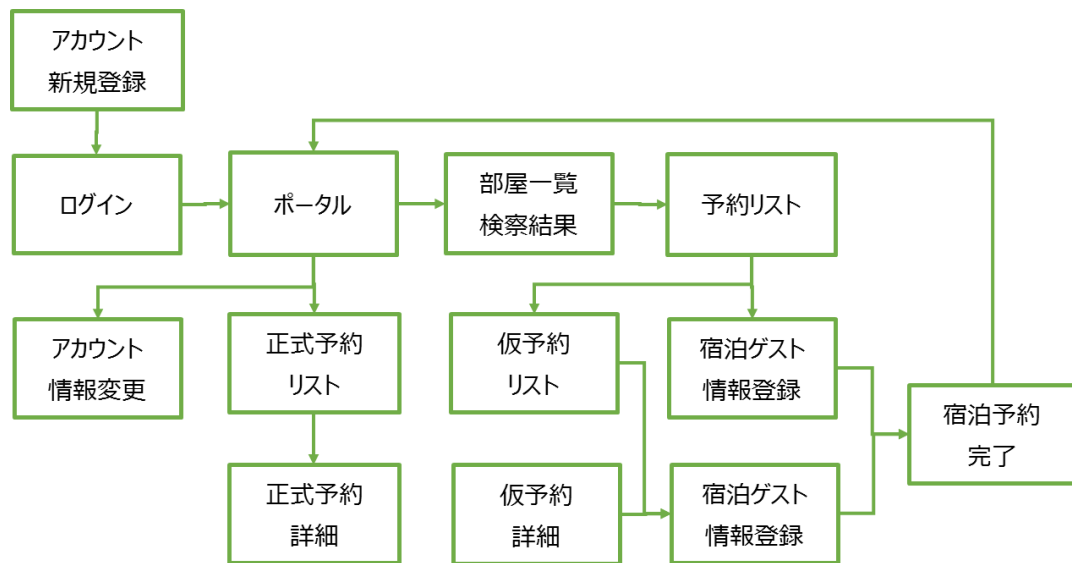


図 4.3.4 B 社の予約システムの機能概要

既存機能を提供する IT システムのアプリケーションは、従来型のモノリシックな構造を持ち、密結合の状態となっており、メンテナンス性に問題があった。そのため、細かな機能の修正を行うにあたっては、影響範囲の確認やテストに時間がかかっていた。また、修正した機能をリリースする際にも、その機能を含んだアプリケーション全体の停止が必要で、リリースに対する制約があり、高い頻度でリリースを行うことが、難しい状態だった。

そうした、状況を変革するため、今回構築するアプリケーションは、マイクロサービスアーキテクチャを活用して、各機能の独立性を高め、それらの間は Web API で接続する構成とした。

また、当案件の特性として、実際の利用者のニーズを見極めながら、提供する機能の優先順位を決定していきたいという意図があった。そのため、開発する機能をフェーズ毎に分けて、段階的なリリースを行うことを、計画していた。

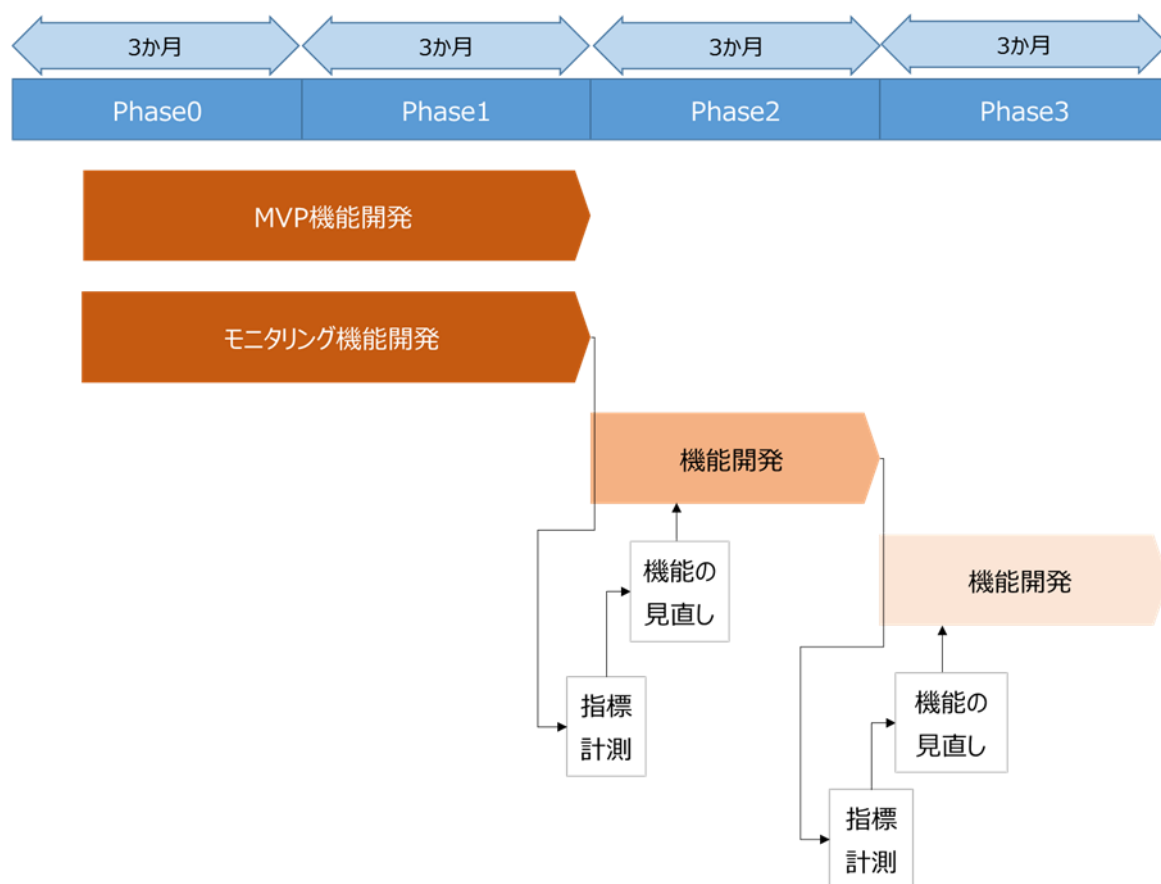


図 4.3.5 B 社の予約システムの開発の進め方

<得られた効果>

①MVP 開発の実現

実用最小限の機能に絞り、それらのリリースを最優先で行った。その後、リリースしたアプリケーションを運用しながら、モニタリングした指標の実績値を元に以降の機能の優先順位を決定し、短期間で機能追加を行った。モニタリングと優先順位の見直しのサイクルを3か月のスパンで、繰り返し行った。

それにより、アプリケーションの利用開始までの期間を圧縮するとともに、リリース済みの機能への影響を最小化しつつ、利用者にとって必要な機能を優先度の高いものから提供することができた。マイクロサービスによる構成を取っていることで、開発する機能の優先順位の入れ替えやリリース時期の変更に、柔軟に対応しながら、開発プロジェクトを完遂することができた。

(3)C 社

<背景と IT システムの概要>

期間は、2020年8月～12月で、機能としては、新たな制度に対する申請の受付機能と自己評価の指標の登録を行う、ITシステムの新規構築を行った。

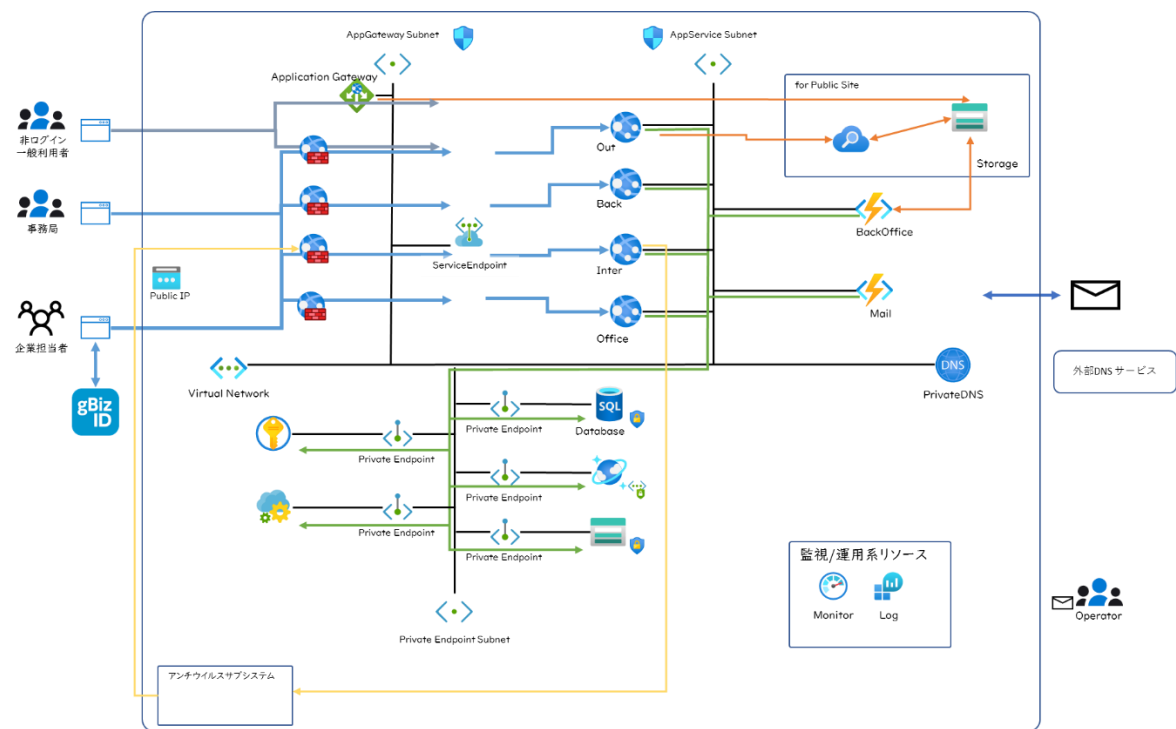


図 4.3.6 C 社のシステム概要図

新たに開始される制度の時期が確定しており、業務の運営のために必須となる機能について、制度開始までに利用者へ提供することが必須であった。そのため、限られた期間で当 IT システムの開発/リリースを行わなければならなかった。ただし、制度自体も本格運用に向けて試行を行っており、その中で、業務面の精査や確認を進めている最中だった。それにより、試行運用中に、当 IT システムに影響する要件変更が発生する可能性が大きいという状態だった。

また、もう一つの機能である自己評価指標の登録は、開始時期が異なっており、制度の受付機能のリリース後に追加する、という段階を踏んで機能を提供するスケジュールで、計画されていた。

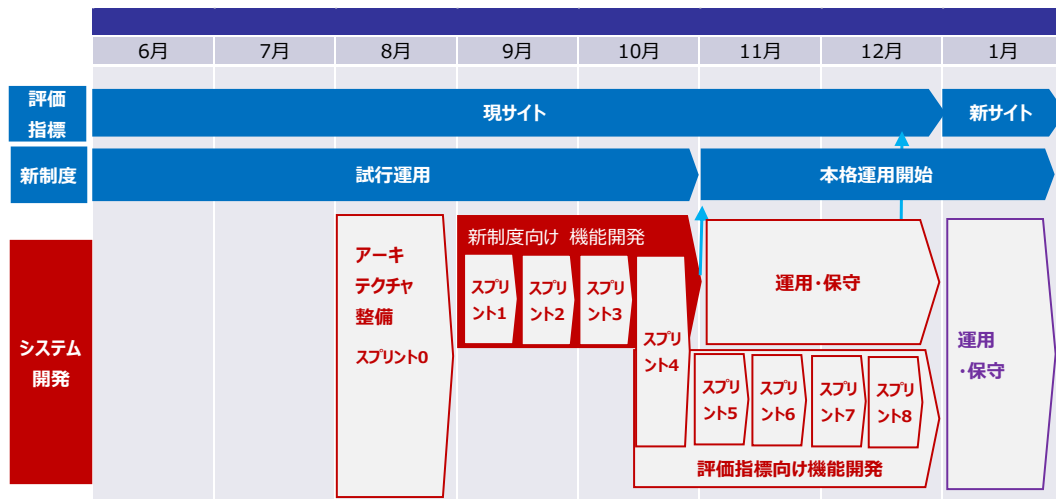


図 4.3.7 新制度、評価指標の開発スケジュール

IT システムのアーキテクチャとしては、段階的なリリース、及び仕様の未確定性へ耐えられるようにマイクロサービスアーキテクチャを採用し、開発手法としてもアジャイル開発を適用した。スプリント毎に、各マイクロサービスの機能に対して、順次、開発を進めていった、また、環境の準備を考慮し、開発、本番環境として、クラウド基盤を活用した。基盤側から提供されるデータストア、ネットワークなどの機能も多数活用し、業務ロジックの実装に集中できるようにした。

<得られた効果>

①短期開発の実現

最初の新制度向け機能のリリースまで、実質 2 か月という短期間で IT システムの開発を実現した。開発プロジェクトの進行と並行して行っている制度の試行から、仕様の変更は実際に発生したが、都度、その仕様改善の取り込みを行いながら、期限通りのリリースを達成することができた。マイクロサービスによる構成のため、仕様変更の影響範囲を限定できたことも、大きな要因となった。

また、次にリリースを行う評価指標の登録機能の開発も並行して進めることができ、全体としての開発期間の圧縮へつなげることができた。

②構成変更への対応

当 IT システムの構築後も、新たなサービスの追加が計画されており、既にリリース済の機能へ影響を与えずに、対応を行うことのできる構成を確保することができていた。

4.4. データ活用

本項では、あるべき IT システムの中のデータ活用基盤(D)・データ分析基盤(E)を構築し、デー

タをビジネスに生かすためのデータ活用について、検討すべき要素や考慮すべき事項について、説明を行う。

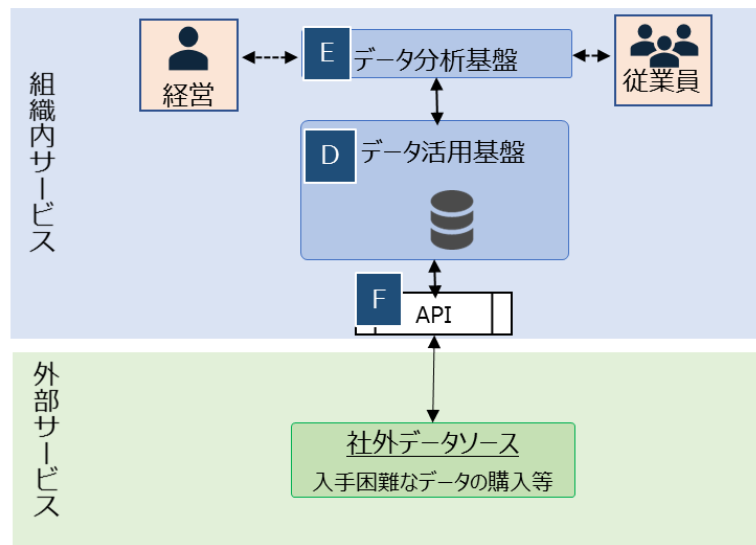


図 4.4.1 あるべき IT システムにおけるデータ活用基盤(D)・データ分析基盤(E)

4.4.1. データ活用のための環境整備のプロセス

4.1 DX を実現するための技術要素群の全体像 でも述べたように、**データ活用基盤(D)・データ分析基盤(E)**を備え、ビジネス上の判断や方針の策定/決定に生かすことは、ビジネス推進上、欠かすことのできない要素となる。

データ活用の目的としては、データを資産として、そこから価値を得られる状態をつくることにある。

企業の中で、通常、データ自体は複数の IT システムに分かれて管理されている。または、必要なデータが、企業の外部にも存在する場合もある。

データの状態についても IT システム毎に様々で、必ずしもそのまま活用できる状態にある訳ではない。

また、企業内にデータがあるからといって、内部の人がそのまま参照できるとは限らない。

そのため、データを活用できるようにするためには、データを複数の場所から集め、かつ状態を整え、必要な人が適切なデータを参照できるような基盤が必要となる。データ活用基盤、およびデータ分析基盤が、その環境にあたる。そして、以下の図のように、データ活用基盤、およびデータ分析基盤を構築し、そこに至るデータの流れを整備することが、データ活用を行う上でのベースとなる。

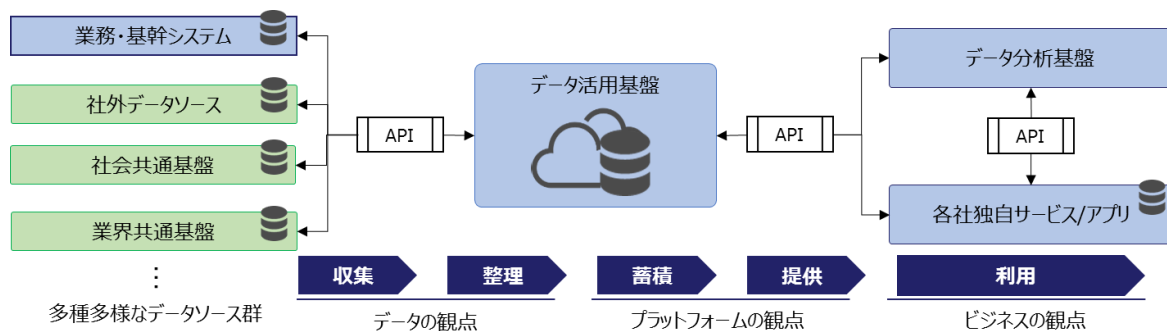


図 4.4.2 データ活用基盤を中心としたデータ活用の流れ

図 4.4.2 の左側にある、社内外の多種多様なデータソース群から、データを収集し、活用できるように整理した上で、データ活用基盤に蓄積する。蓄積したデータを右側のデータ分析基盤や各社の独自サービス/アプリから利用するという流れとなる。そして、データソース群とデータ活用基盤との間、データ活用基盤とデータ分析基盤等の間は、API をインタフェースとして、連携するという構成となっている。

そうした環境を整えるプロセスとしては、大きく以下のような流れとなる。

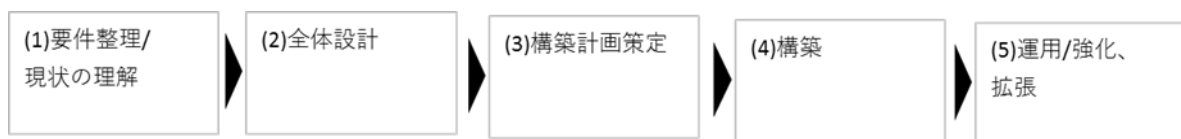


図 4.4.3 データ活用環境整備のプロセス

(1)要件整理/現状理解

(1)においては、データ活用環境に必要となる要件の整理、およびデータの現状の理解を行う。

要件整理としては、データ活用にどういうデータが必要で、そのデータの鮮度、精度、粒度、頻度、範囲、量がどういったものであるか整理を行う。

そして、現状の理解として、そうしたデータが、どの IT システムに、どのような状態で存在するかを、調べて理解し、整理する。

(2)全体設計

(1)で整理した要件を満たすように、全体設計として、データ活用基盤/データ分析基盤のアーキテクチャ(データアーキテクチャ)のデザイン、および必要なデータをデータ活用基盤に集めるためのデータフローの設計を行う。

データフローとしては、データを保持している IT システムから、抽出し、変換した上で、データ活用基盤へ取込を行う一連の流れに対して、全体での設計を行う。

IT システムの全体像とともに、データの利用方針、権限など、統制に関わる部分(データガバナンス)についても、検討する。

(3)構築計画策定

全体設計で、全体像を描いても、一度にすべてを構築するには、時間やコストが必要になることもあるので、計画を策定する上で、実現すべき要件に対して、構築の対象範囲の明確化やそれらの優先順位付けを行う。

その優先順位に従って、全体像の構築に向けたロードマップを描き、構築計画の策定を行う。

(4)構築

(3)で策定した計画に従って、データ活用基盤/データ分析基盤の構築を行う。

構築に際しては、データ活用基盤に格納する、データの構成(データモデリングとデザイン)を確定するとともに、そのデータを管理している IT システムからのデータフローの処理内容や運用方法(データ統合と相互運用性)を決める。また、データ自体にもトランザクションデータやマスタデータなど種類があるが、その中でもマスタデータは、データを活用する際の軸となるデータとなるため、変更時の対応などを含めそれらの管理(参照データ管理、マスタデータ管理)について明確にする。

それ以外に、データを利用する際に必要となってくる、データに関する情報(いつの時点のデータか、出所はどこかなど)の管理(メタデータ管理)についても、必要に応じて、検討する。

また、上記の運用や管理に関して、品質の維持(データ品質管理)についても方針を決め、計画を策定する。

(5)運用/強化、拡張

構築後は、品質を確保しつつ運用を行い、データソースの修正などに対応していく。

また、扱うデータの対象範囲を広げる、新たな分析など機能の追加を進め、データ活用の有効性をより高めていく。

4.4.2. データ活用を進めるために必要となる要素

データ活用を進め、有効なものとしていくためには、企業の中で、統制を取り、適正な利用を行い、品質を保ったデータを扱い、効率的に運用していくことが、重要となる。そのために、データマネジメントの実践が、必要とされることになる。

データマネジメントとして、どのような内容がそこに含まれるかを、見ていきたい。

データマネジメントに関する事項について包括的に記載されている資料としては、DMBOK2⁵⁶が広く使われている。そこで、本項では、DMBOK2 に則った形で、データマネジメントに関する説明を行う。

⁵⁶ DAMA International(編著)、DAMA 日本支部、Metafind コンサルティング株式会社(監訳)『データマネジメント知識体系ガイド (Data Management Body Of Knowledge) 第二版』日経 BP 社の略

DMBOK2 におけるデータマネジメント機能のフレームワークは、以下の図の通りである。

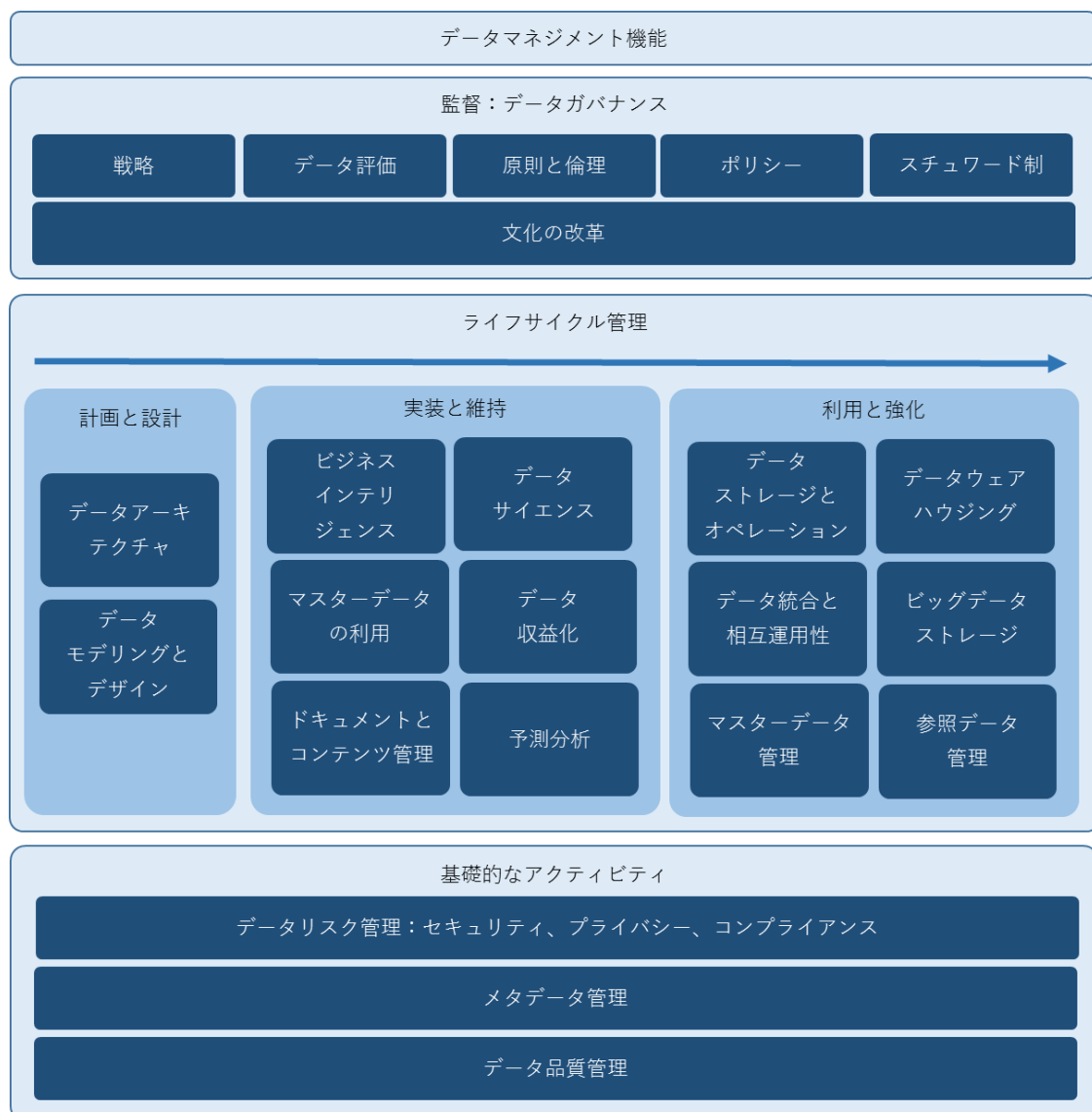


図 4.4.4 データマネジメント機能フレームワーク

ここにおいて、データマネジメント全体に関わる機能としてまずデータガバナンスがおかれる。これを基に、実際の実働要素であるデータのライフサイクル管理がおかれ、ここにはデータに関するアーキテクチャ設計、分析等利活用機能の実装、それらによるデータ統合と相互運用等、現実の利活用などの諸機能に関わる。そして、そうした業務を支える基礎的な要素として、メタデータ管理やデータ品質管理が存在する。

ここでは、上記のフレームワークで、主要な機能と考えられる以下の要素について、記載する。(各機能の詳細については、DMBOK2 を参照されたい。)

- (1) データガバナンス
- (2) データアーキテクチャ
- (3) データモデリングとデザイン
- (4) データ統合と相互運用性

(5)参照データ管理、マスタデータ管理

(6)メタデータ管理

(7)データ品質管理

(1)データガバナンス

データガバナンスとは、DMBOK2では「データ資産の管理に対して職務権限を通じ統制すること」と定義され、データガバナンスの目的は「データに関するポリシーとベストプラクティスに従ってデータを適切に管理すること」とされている。

<データガバナンスの意義>

DXやデータドリブンな経営を進める上で、データは非常に重要な資産である。その資産を守ったり、ビジネスに活用したりするために、データマネジメントに関するポリシーやデータ戦略を策定することが必要になる。

(2)データアーキテクチャ

データアーキテクチャは、DMBOK2では「システムの現状、一連のシステムの構成要素、システム設計の規範（アーキテクチャの実践）、単独またはシステム群の意図的な設計、システムが記述された制作物（アーキテクチャドキュメント）、設計作業を行うチーム（アーキテクトやアーキテクチャチーム）」を指すとされている。また、データアーキテクチャは「企業のデータニーズを明確にし、ニーズに合うマスタとなる青写真を設計し、維持する。マスタとなる青写真を使ってデータ統合を手引きし、データ資産をコントロールし、ビジネス戦略に合わせてデータへの投資を行う。」と定義されている。

<データアーキテクチャの意義>

デジタル化やDXが加速していく中で、部分最適でITシステムを導入したため、ITシステムがつぎはぎ状態になっている企業もある。そのような状態ではデータを十分に管理、利活用することが難しい。そのため、ビジネス戦略やデータ要件などのインプットに基づき、ビジネスを俊敏に展開できる最適なアーキテクチャを設計する必要がある。

(3)データモデリングとデザイン

データモデリングとは、DMBOK2では「データ要件を洗い出し、分析し、取扱いスコープを決めるプロセスである」と定義されている。また、洗い出したデータ要件は、データモデルに表現する。

<データモデリングとデザインの意義>

データ要件を洗い出し、データモデルを作成することで、データ資産を視覚化し、把握することができる。また、適切なデータモデリングを行うことで、データの重複管理などを避けることができ、運用コストの削減や将来の新しいアプリケーションの構築コストの削減も期待できる。

(4)データ統合と相互運用性

データ統合と相互運用性は、DMBOK2 上では「データストア、アプリケーション、組織などの内部とそれらの相互間で実行される、データの移動と統合に関するプロセス」と定義される。なおデータ統合は「データを物理的、仮想的を問わず、一貫した形式に統一すること」、データの相互運用性は、「様々なシステムがどの程度情報を連携できるかを表すもの」とされる。

<データ統合と相互運用性の意義>

IT システム間、組織間におけるデータ統合の推進や相互運用性を高めることで、以下の効果が期待できる。

① 利便性の向上

同じ組織でも、部署毎に IT システムが縦割りで、情報がサイロ化しており、実は類似した調査や研究、開発を行っていたということや、同じお客様に複数人の営業がアプローチしていたといったことはよく聞かれる。IT システムや組織の垣根を越えて、データ統合、相互運用を行うことで、利便性が向上し、そのような悩みを解決することができる。

② 全体最適化とコスト削減

データを統合することによって、類似データの整理、集約が図られ、データの重複管理がなくなり、全体最適化を図ることができる。またビジネス上のコスト削減を考えた際にも、単体の業務で削減できることは限られている。複数業務のデータを統合することで、業務の効率化を図ることもできる。

③ 新しい価値の創出

IT システムや組織間をまたいでデータを連携、統合していくことで、新たな知見が創出できることもある。

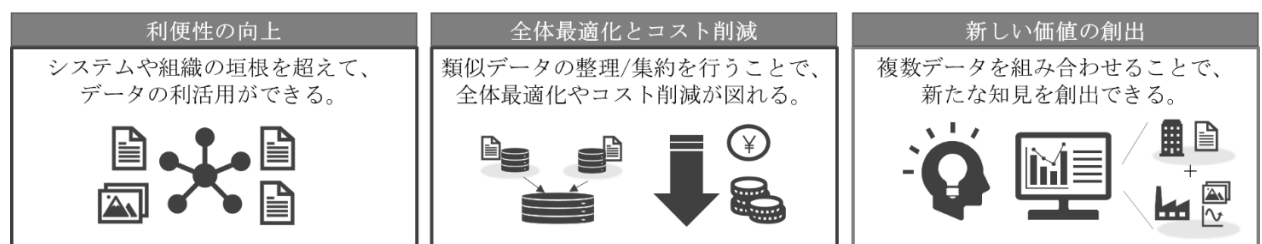


図 4.4.5 データ統合や相互運用の意義

(5)参照データ管理、マスタデータ管理

参照データとは、「他のデータを特徴付けたり、データベース内のデータと外部組織の情報とを関連付けたりするために使用されるデータである」と DMBOK2 で定義されており、最も基本的な参照データはコードと摘要で構成されているが、マッピングや階層を持つより複雑なものもあると説明している。

同様にマスタデータは、「業務活動に関連する共通概念を抽象的に表現することにより、そ

の活動に意味を与える。顧客、製品、従業員、ベンダー、制御ドメイン(コード値)など、組織内部と外部で業務上の取引に関わる実体を記述した詳細」とDMBOK2で定義されている。

<参照データ管理、マスタデータ管理の意義>

参照データを管理する意義は「運用効率とデータ品質の向上である」とされており、複数の業務部門が独自のデータセットを維持するよりも、参照データを一元的に管理の方がコスト効率は向上する。また、ITシステム間で矛盾が起きるリスクも低減する、とDMBOK2では説明されている。更にDMBOK2では、「参照データ管理では、ドメイン値定義、項目定義、ドメイン値内とドメイン間の関係などを制御し保守することになる。参照データ管理のゴールは、異なる機能間で値の一貫性と最新性を保証し、組織がデータにアクセスできるよう保証することである。他のデータと同様に、参照データにはメタデータが必要である。参照データの重要なメタデータ属性には、その取得源、例えば業界標準の参照データの統治団体(いわゆる業界団体)などが含まれる。」とされている。

また、DMBOK2では「マスタデータは主要業務の対象であり、組織全体の様々なアプリケーションで使用され、そこには関連するメタデータ、属性、定義、役割、関連性、タクソノミ⁵⁷などが含まれる」と説明され、更に「マスタデータの対象となるのは、組織にとって最も重要な「モノ」であり、トランザクションとしての中に記録され、報告され、測定され、分析されるデータである。」とされている。

(6)メタデータ管理

メタデータは、「データに関するデータ」である。メタデータとして分類できる情報の種類は幅広く、DMBOK2上では、「ITプロセスと業務プロセス、データのルールと制約、論理的および論理的なデータ構造、以上に関する情報がメタデータに含まれる」とされる。

また、メタデータが活用されるようになった背景としては、以下の点が挙げられる。

近年デジタル化が進み、データ量が膨大になっている。データがデジタル化されて、1つのITシステムに集約されていたとしても、何のデータか一目でわからなくては、データの利活用が進まない。紙のデータである本が大量に所蔵される図書館にも図書目録があるが、デジタルデータに関しても、データを探すための辞書の役割を果たすメタデータが必要と言える。

⁵⁷ 概念やデータなどを上位／下位の関係に整理した階層構造。

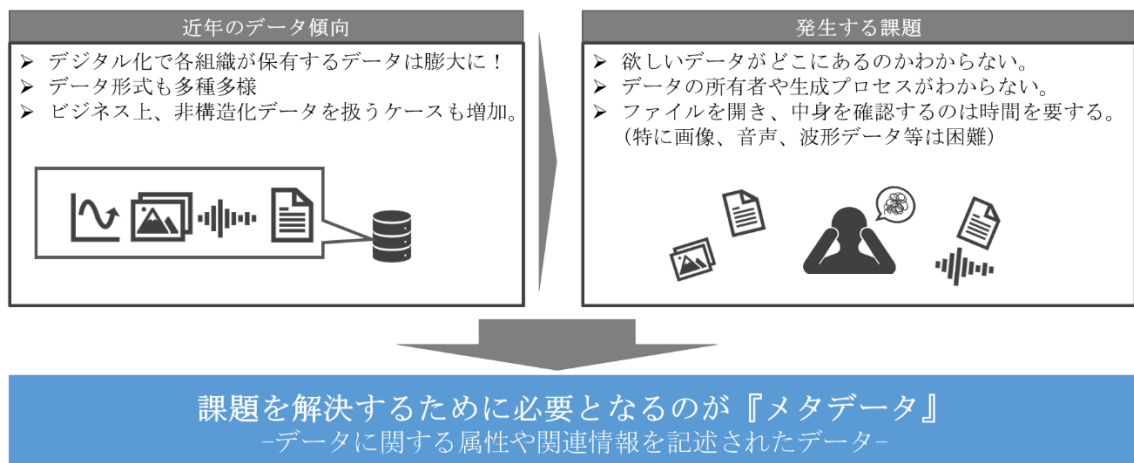


図 4.4.6 メタデータの策定背景

<メタデータ管理の意義>

メタデータを管理し、利用する意義としては、以下3点が挙げられる。

① データの検索の容易化

データが一か所に集約されていたとしても、欲しい情報がどこにあるか検索できず、データの管理者に問い合わせることは少なくない。メタデータはデータに関するデータ、つまりデータに関する属性や関連情報を記述したものであり、その課題を解決する役割を果たす。データ利用者は、独力で欲しい情報を探し出すことができるようになる。またデータ管理者も、利用者からのデータに関する問い合わせ対応の工数を削減することができる。

本調査のヒアリング事例では、グループ内データプラットフォーム基盤において、データカタログを設け、アクセス権があるグループ会社であれば、何のデータを保有しているかわかる仕組みを設けられていた。

② 様々な観点でデータを分析、可視化

メタデータをもとに、データが整理できるようになるため、分析、可視化も容易になる。

例えば、SNSへの投稿画像を例にすると、作成者、場所、写っている画像の分類がメタデータとして付与されていたら、その情報毎にデータを整理し、分析、可視化することが可能となる。様々な観点でデータを俯瞰できることから、新たな知見を創出するきっかけにも繋がる。

③ データ連携の促進

近年、組織を超えたデータ連携が盛んになっている。しかし組織間のデータを統合しても、お互いのデータが何を意味しているのかがわからなければ、その後のデータ利活用は進まない。その点、メタデータはデータが何を表しているのかを示すものであり、付与することで、データ連携促進の一助となる。



図 4.4.7 メタデータの意義

(7)データ品質管理

データ品質は、DMBOK2では「データ利用者の期待と要求を満たす度合い」と定義されている。データが果たすべき目的に足るものであれば品質は良いとされ、目的に足らない状態であれば低品質とされる。

<データ品質管理の意義>

低品質のデータを利用すると、データクレンジングにコストを要し、データを利用して提供されるサービスやデータ分析結果の品質にも悪影響を及ぼすため、データ品質は非常に重要な要素で、維持や向上の取り組みは不可欠である。一方でデータ品質をあげようとするコストがかかる。そのため、適切な品質レベルを設定する必要がある。

4.4.3. 要素ごとの考慮点

前節で説明した(1)データガバナンスから(7)データ品質管理までのデータマネジメントを構成する主要な要素に関して、検討する上で考慮すべき事項について、要素ごとに説明する。

(1)データガバナンス

データガバナンスについて、データマネジメント全体の方針に関連しうる事項に関して、考慮点を挙げる。

① ビジネスへのデータ利活用

ビジネスへのデータ利活用のアプローチとしては大きく2つ考えられる。

データやデータ分析を起点に新ビジネス創出や業務改善を行うアプローチと、ビジネスを起点に、新ビジネス創出や業務改善の課題に対して、必要なデータを集めるアプローチである。

➤ データ起点

データ起点のアプローチでは、データを業務の課題解決にどう活用できるか、具体化されるところまではできていないこともある。その意味でも、データをビジネスに結びつけるためにはビジネス部門の参画が不可欠である。また、まずはデータを可視化することで着想を得るなどの工夫が必要となる。

➤ ビジネス起点

ビジネス起点のアプローチでは、必要となるデータがない、あるいは整理されていないということもある。データ収集に際しては、経営層やビジネス部門の協力を仰ぐ必要があるが、その際にアウトプットのイメージを提示することで、データ収集の目的を共有でき、推進し易くなる。まずは小さな成果を出すことを目指し、スモールスタートで進めることも、着実に DX を進めるためには有用となる。

表 4.4.1 データ利活用のアプローチ

データ利活用のアプローチ	課題	工夫
データ起点	✓ データを業務の課題解決にどう活用できるか、具体化されるところまでできていない	✓ ビジネス部門の参画 ✓ データの可視化からの着想
ビジネス起点	✓ ステークホルダを巻き込んだデータ収集や整理が必要になる	✓ アウトプットのイメージをステークホルダに提示 ✓ スモールスタートで進める

② セキュリティ

4.1.4(2)に記載したように、データ資産保護のためのセキュリティは、データの種別によらず、機密性のレベル分け、及びそれに基づくアクセス権設定・アクセス制御が基本である。またデータを分析・利活用する場合、その目的・方法の妥当性について、4.1.4(4)に記載したようなプライバシー・コンプライアンス関連規定の遵守が求められる。以下では、これらのセキュリティ・プライバシー・コンプライアンス施策を実施するための考慮点をあげる。

➤ 体制

データ管理に関係する社内部門としては、情報システム部門、情報セキュリティ部門の他、知財・法務等のコンプライアンス部門、実際にデータを使ってビジネスを行う事業部門等がある。また近年、データ利活用や DX 等の進展に伴い、データドリブンな事業を志向する企業では、これらの部門を横断的に連携させ、かつ関連する部門全体を横串で調整する統括者として最高データ責任者（CDO:Chief Data Officer）をおくことがある。最高データ責任者は、同時に DX の統括責任者（CDO:Chief Digital Officer）である可能性も高いと考えられる。

すなわち、データセキュリティに関する体制は情報システム・セキュリティ部門に閉じたものでなく、誰が何の目的でデータを利用するのかを明確化する部門、及び利用の正当性（外部調達データに契約上の制限はないか等）をチェックするコンプライアンス部門等が連携できる体制、およびこれら部門間の意見を調整し、共通のアクセスルール（共有ルール）を策定・実施できる統括者（CDO）が必要である。

➤ データ保護とデータ利活用のギャップ調整

データ保護とデータ利活用の調整が難しくなることがある。例えば、従来別々の事業で利用していたデータを統合分析して新たなサービスを創る、という事業計画があったとしても、これまでの事業責任分担・データアクセス権が縦割りであり、機密分類レベルも未統合、用語や書式も異なる、等の理由でうまく進まないケースが考えられる。こうしたケースに即座に対処することは難しいが、上記の体制においてパイロットプロジェクトを起こし、統合分析可能なデータを新たに作成してもらってあるべき連携を試行する、等のアプローチも考えられる。

➤ 外部サービス・サプライチェーンのセキュリティ・ガバナンス

データ収集・分析においても外部サービスの利用が想定される。収集では、業務委託、他企業からの購入等があり得る。分析では、その一部を SaaS サービス、他企業等に委託するケース等があり得る。しかし、IT 業務委託契約においてセキュリティ要件を具体的に記載し、それをきちんとチェックすることは十分行われていない。契約時にセキュリティ要件を詳細化できない、委託先のセキュリティ状況チェックが高コストになる、等の原因が考えられ、サプライチェーン上のセキュリティ対策状況チェックは大きな課題となっている。

こうした中、業務委託先・調達先を含む外部サービスが然るべきセキュリティ対策を行っているかを確認する方法としては、公開情報、例えば当該サービスの事業者が取得しているセキュリティ認証 (ISMS, P マーク等)、準拠しているセキュリティ基準 (PCI DSS 等)、公的機関が公開する事業者リスト (ISMAP 等) への登録、公開されているセキュリティポリシー、等を確認することで一定の評価が可能である。また、契約前にセキュリティ対策に関するチェックシートを提出してもらい、評価に用いる組織もある。外部サービスの選定にあたっては、上記のような方法で確認を行い、不安な点は対応を確認したうえで契約する、等によりセキュリティを確保することが望ましい。

③ ビジネスのグローバル化

グローバル化するビジネスの中で、各国法や条例の順守、変化への対応にも配慮しなければならない。国や地域をまたぐデータの収集やその扱いなどもその範疇となり、組織的な対応が必要と考えられる。

(2) データアーキテクチャ

データアーキテクチャについて、アーキテクチャの設計、構築の際に、ポイントとなる事項に関して、考慮点を挙げる。

① アーキテクチャ設計におけるデータ要件

➤ リアルタイム性

リアルタイム性が求められるデータと求められないデータでは、アーキテクチャの設計は異なるので、要件を見極め、全体像を描いていく必要がある。

➤ 拡張性

DX の初期段階では、利用するデータ量やデータ項目が確定していないことが多い。そのため、データ容量、データ項目が増えることを想定した構成にする必要がある。

そうした点で、拡張性の高いクラウド環境が使われる場合も多い。

➤ 非構造化データ

画像、音声、センサーデータなどの非構造化データも増えている。非構造化データの場合、バイナリデータを RDB (Relational Database) に格納し、メタデータでデータの管理を行い、データを検索する方法や、NoSQL (Not Only SQL) にデータを格納する方法がある。

✓ RDB への格納

RDB ではデータ処理が一貫している点、トランザクション処理を行うことができる点がメリットであるが、膨大なデータ量を扱う場合は処理速度が落ちてしまうなどのデメリットがある。

✓ NoSQL への格納

NoSQL で大量データの処理でも処理速度が落ちないメリットがあるが、データの整合性の観点は結果整合性のため、RDB と比較し一貫性が緩いというデメリットがある。

② アーキテクチャの代表的な構成

➤ 3 層構造

データ分析基盤としては、データレイクの Raw, Curated, Published Zone からなる 3 層構造が基本的な構成である (図 4.4.8 参照)。これをベースに企業毎にカスタマイズしていくことが多い。

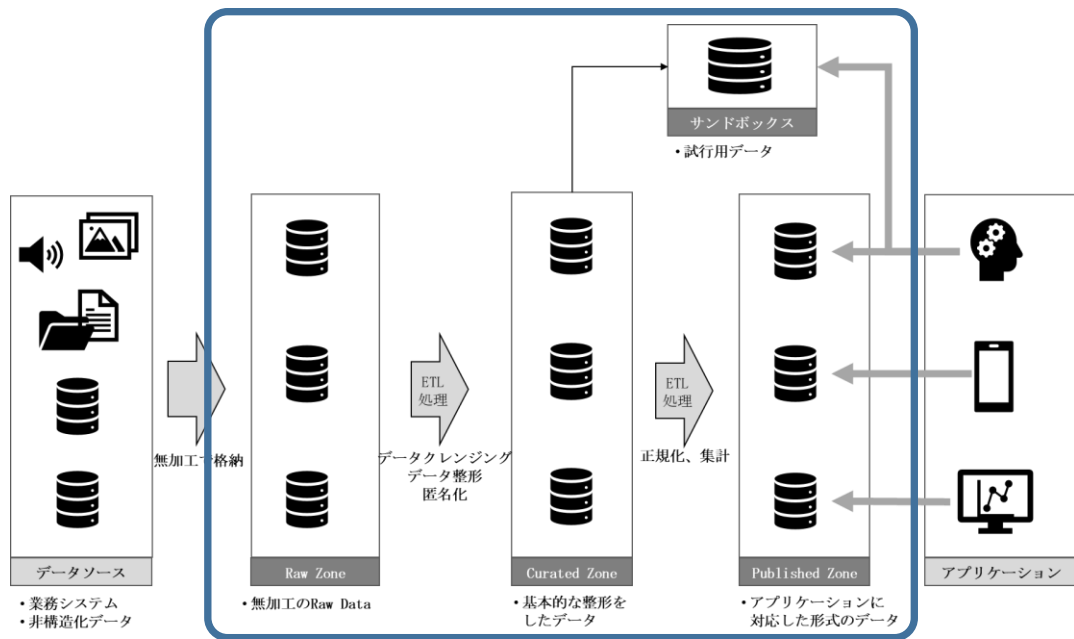


図 4.4.8 3層構造アーキテクチャのイメージ

✓ Raw Zone

Raw Zone には、各データソースから送られてきたデータを無加工の Raw Data として配置する。Raw Data を残しておくことで、IT システム仕様や分析観点が変わった場合でもデータを利用することができる。

✓ Curated Zone

Curated Zone には、Raw Zone のデータを扱いやすいように整形したデータを配置する。フォーマットの変換や匿名加工などもこの層で行う。Curated Zone があることで、データの結合や、View の作成が容易にできる。Curated Zone のデータをトライアル環境としてサンドボックスに格納することもある。

✓ Published Zone

Published Zone には、Curated Zone のデータを正規化や、集計しアプリケーションに対応した形式にしてデータを配置する。場合によっては Curated Zone のデータを組み合わせて新たなテーブルを作成することもある。Published Zone のデータは分析ツールやアプリケーションから利用でき、検索が可能である。

➤ 仮想データベース

近年、仮想データベースを利用するケースも増えてきつつある。仮想データベースとは、複数のデータソースのデータに対して仮想ビューを通してアクセスし、物理的にデータを蓄積しないデータベースである（図 4.4.9 参照）。仮想データベースのメリットとしては、物理的にデータをコピーする必要がないこと、データをリアルタイムに最新の情報にすることが可能なことが挙げられる。

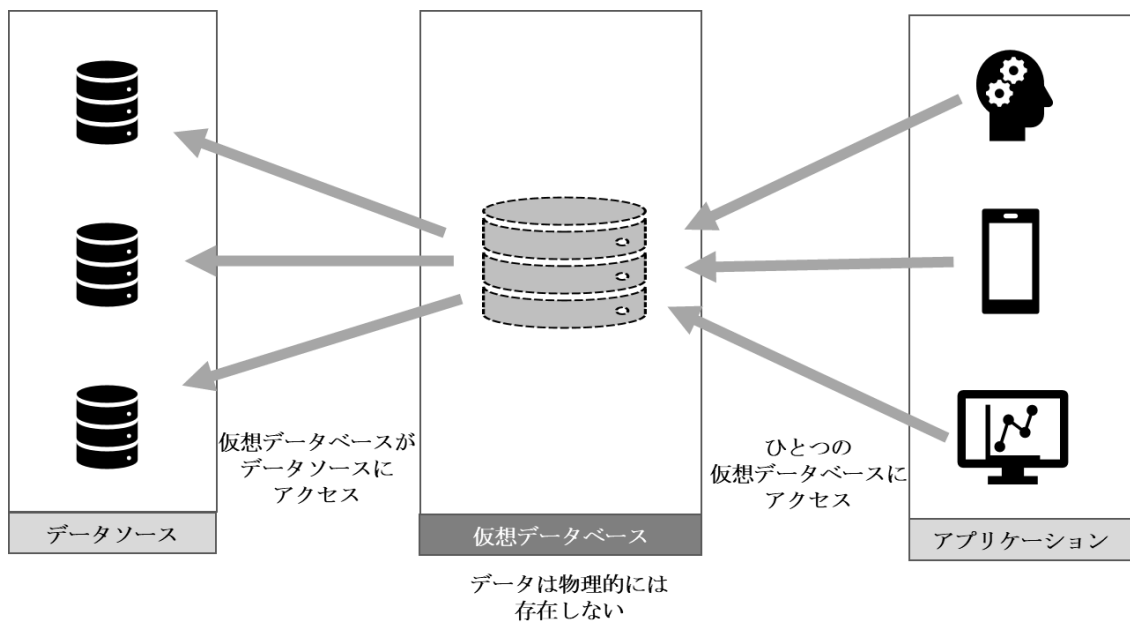


図 4.4.9 仮想データベースのイメージ

③ アーキテクチャ構築の進め方

アーキテクチャ構築においては、段階的に行うことがスピード感のある DX には求められる。

➤ 初期段階

初期段階では、データ要件やデータモデルが確定していない場合が多い。アジャイル型で試行錯誤しながらアーキテクチャ設計と評価を繰り返し行うことで、修正が必要になった際でも容易に修正ができ、柔軟な対応ができる。

➤ 成熟段階

初期段階でデータ要件、データモデルが確定し、最適なアーキテクチャの構築ができた際には、設計書やデータ活用プロセスなどを文書化し、データ資産を可視化、継続的にメンテナンスができるようにする必要がある。

(3) データモデリングとデザイン

データモデリングとデザインについて、推進する上で、課題となることが多い事項に関して、考慮点を挙げる。

① データ収集とビジネス観点でのデータの理解

データ利活用のためのデータモデリングの第 1 歩としては、現在あるデータを収集し、その実態を把握することが必要である。その後にデータ利活用のためにデータモ

デルを構築する。情報システムがサイロ化し、そこに蓄積されているデータもサイロ化され、どのようなデータがあるか把握できていない場合も多い。また、似たようなデータ項目が複数存在するという課題もある。

複数の IT システムに散在していたデータを収集し、データの分類、モデリングに際しては、ビジネスの観点でデータの理解が不可欠である。また、業務に利用する場合、業務プロセスとの整合性を取る必要があり、ビジネスとのつながりは必要になる。

② 推進体制

データの整理に際しては、ビジネス部門の現場でデータの意味合いや使用方法が属人化されているものもあるため、データの正しい理解のためにはビジネス部門の協力が不可欠である。更に、データとビジネスの両面でモデリングを進めるためにはデータエンジニアのような専門的な人材、組織が必要になる場合もある。

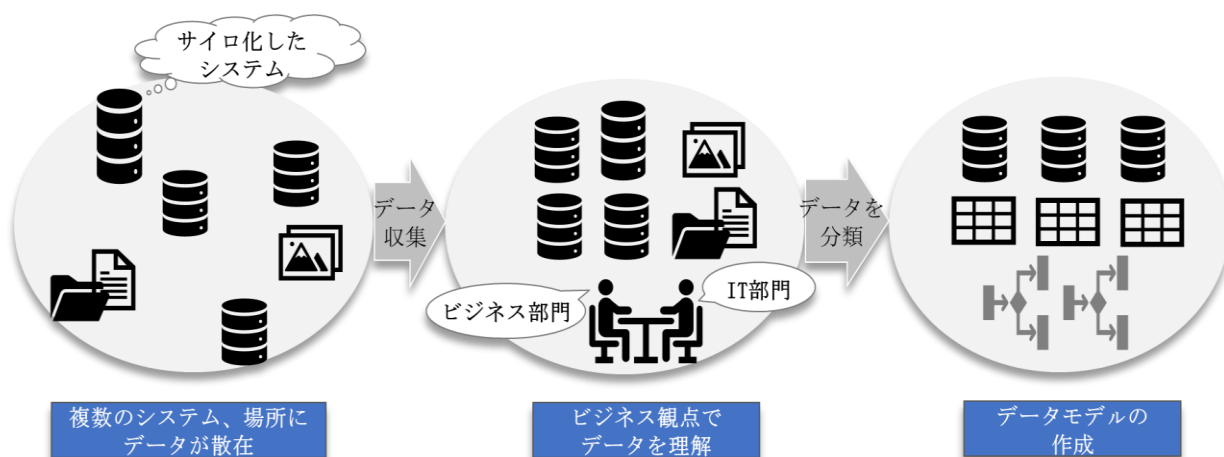


図 4.4.10 データモデリングの進め方

③ 汎用サービスやパッケージの利用

従来は自社の業務に合わせてオーダーメイドで IT システムを構築することが主流であったが、現在は外部の汎用サービスやパッケージの利用も増えている。汎用サービス、パッケージの開発ベンダーは業界、業務の標準的なデータモデルを作成し、自社製品に組み込んでいることが多い。SaaS などの汎用サービスやパッケージなどで定義された標準的なデータモデリングに合わせ、そのパッケージのカスタマイズにより、新しい業務を展開していくことが、スピート感のある DX のポイントになる。

④ モデルの再生成、管理

以前は市場やユーザーは簡単には変化しないという前提があり、ひとたびデータから法則性やモデルを生成してしまえば、ずっとそれが使えるというのが常識だった。

ところが DX 時代になり、市場やユーザーは不確定であいまいで変わりやすいものと

なってしまった。つまりデータから法則性やモデルを生成しても長期的には現実との差が開いていき、日々精度が下がっていくこととなる。

つまり市場やユーザーの変化、それによるデータの変化によって、法則やモデルも変化していくのでモデルの再生成や管理も重要な要素となる。

(4) データ統合と相互運用性

データ統合と相互運用性について、その方式等を検討していく上で観点となる事項について、考慮点を挙げる。

① データ統合、相互運用の方式と優先度の検討

事業所の統合を行うような場合、使用する IT システムが異なるので、データをどのように統合するかという議論が浮上する。この場合、コストとビジネスのバランスを踏まえて、方式を決めることが重要となる。以下の表のように、大きく 3 つの方式が考えられる

表 4.4.2 データ統合や相互運用の方式

#	方式	内容
1	既存システム間のデータ連携	IT システム間で API 等を介してデータをやり取りし、相互運用ができるようにする。
2	既存システム間のデータ統合	同じ内容を指すデータが、異なる ID や用語で定義されている場合、名寄せした上でデータを統合する。
3	新規システムの構築	新しい IT システムを一から作り、そこに統合する。

② データ要件の検討

データ統合や相互運用上、データの要件は様々ある。その中でも特に「データ項目の定義」と「データ連携の頻度やファイルサイズ」は重要な要素である。

「データ項目の定義」は、同じ内容を示していても、単位が揃っていない場合もある。誰もがわかる定義というものが求められる。「データ連携の頻度やファイルサイズ」も重要である。例えば、リアルタイムに連携をしないといけないものであれば、バッチではなく、API で連携する必要がある。またその際のファイルサイズが大きければ、通信回線も検討する必要がある。

③ データのアクセス権の策定

データの統合や相互運用を図る上では、データのアクセス権の検討も欠かせない。データのアクセス権は、実現したいビジネスや、データの利用規約に応じて異なるため、データに関わる関係者と十分協議した上で、アクセス権を定めていく必要がある。

(5) 参照データ管理、マスタデータ管理

参照データ、マスタデータ管理について、効果的な管理を実現するため、推進に関する考慮点を挙げる。

① 推進体制

マスタデータの定義や運用を考える上で推進体制は重要である。

マスタデータによっては、複数の IT システムが関連しているケースもあり、かつ、それぞれが運用部門を持っていることもある。そうした場合、マスタデータ管理の全体を把握する必要があり、全体を統括するために、運用部門を束ねる組織の検討も必要となる。

② DX 推進に必要なマスタデータの考え方

マスタデータは、通常、企業内で一元管理されることで、品質を保ち、有効な活用へつながることになる。ただし、以下のようなマスタデータの管理の仕方も、実践的な観点となる。

DX の推進に必要とされるスピードを実現するために、業務を限定し、特定のマスタデータについては、一元管理せず、分けて管理すると共に、外部サービスの活用を優先することも検討する。

(6) メタデータ管理

メタデータ管理について、策定からメンテナンスまでの各フェーズにおける考え方に関して、考慮点を挙げる。

① メタデータの策定

メタデータの策定は、業務知識を必要とする上、業務関係者が多く、共通理解を図るには時間を要する。従って、いかに効率的に業務関係者の認識共有を図るかが重要となる。例えば、外部のサービスが提供している実績のあるテンプレートがある場合は、それを活用することも一つの手である。

② メタデータ管理の効率的な運用

既存の膨大なデータに対し、メタデータを付与するのは困難である。また新規データであっても、手入力でミスなく、メタデータを付与することは労力を要する。従って、入力が容易になるようなインタフェースの採用や AI の活用等、メタデータを効率的に付与できる仕組みの検討も必要である。

③ メタデータのメンテナンス

データ自体に変更が入ると、それに伴い、メタデータの変更も必要な場合がある。従って、変更の可能性やリスクを洗い出し、事前に変更プロセスや体制を整備しておくことが必要だ。

(7) データ品質管理

データ品質管理について、その有効性を高めるための観点に関して、考慮点を挙げる。

① ビジネス戦略に合わせたデータ品質の評価軸

データ品質はデータ利用者からの要求によって設定される。ISO/IEC 25012 のデータ品質の評価軸を表 4.4.3 に示す。ビジネス戦略やデータ利活用の目的に照らし、必要となるデータ品質の評価軸、品質のレベルを決め、データあるいは IT システムでの対応を講じる必要がある。

表 4.4.3 データ品質の評価軸

データ品質の評価軸	概要	データ 依存	システム 依存
正確性	特定の利用状況において意図した概念又は事象の属性の真の値を正しく表現する属性をデータがもつ度合い	○	
完全性	実体に関連する対象データが、特定の利用状況において、全ての期待された属性及び関係する実体インスタンスに対する値をもつ度合い	○	
一貫性	特定の利用状況において矛盾がないという属性及び他のデータと首尾一貫しているという属性をデータがもつ度合い	○	
信ぴょう性	特定の利用状況において利用者によって真（実）で信頼できるとみなされる属性をデータがもつ度合い	○	
最新性	特定の利用状況においてデータが最新の値である属性をもつ度合い	○	
アクセシビリティ	幾つかの障害が原因で支援技術又は特別の機器構成を必要とする人々が、特定の利用状況において、データにアクセスできる度合い	○	○
標準適合性	特定の利用状況において、データの品質に係る規格、協定又は規範及び類似の規則を遵守する属性をデータがもつ度合い	○	○
機密性	特定の利用状況において、承認された利用者によってだけ利用でき、解釈できることを保証する属性をデータがもつ度合い	○	○
可用性	特定の利用状況において承認された利用者及び／又はアプリケーションがデータを検索できる属性をデータがもつ度合い	○	○
効率性	特定の利用状況において、適切な量及び種類の資源を使用することによって処理することができ、期待された水準の性能を提供できる属性をデータがもつ度合い	○	○
精度	正確な属性、又は特定の利用状況において弁別を提供する属性をデータがもつ度合い	○	○

追跡可能性	特定の利用状況において、データへのアクセス及びデータに実施された変更の監査証跡を提供する属性をデータがもつ度合い	○	○
理解性	利用者がデータを読み、説明することができる属性で、特定の利用状況において、適切な言語、シンボル及び単位で表現された属性をデータがもつ度合い	○	○
移植性	特定の利用状況において既存の品質を維持しながらデータを一つのシステムから他のシステムに実装したり、置き換えたり、移動したりできる属性をデータがもつ度合い		○
回復性	特定の利用状況において故障発生の場合でさえ、明示された水準の操作及び品質を継続し、維持することを可能にする属性をデータがもつ度合い		○

(資料：JIS X 25012:2013 ソフトウェア製品の品質要求及び評価を基に作成)

② データクレンジング

多くの企業で、データ利活用の際にデータクレンジングを行っている。

業務上の知見や既存システムの知見がないと扱うのが難しいデータがあり、自社でノウハウを蓄積し、クレンジングを実施するなどの対策が必要となるケースもある。

データクレンジングを機械的に実施できるツールも存在している。そういったツールを活用して、データクレンジングを進めることも有用である。ただ、誤変換が生じる可能性もあるため、十分な検証を行った上での利用が賢明となる。

③ 推進体制

データ品質は、継続的に監視、評価、改善を行うことが重要となる。製品やサービスの品質を担保するために品質保証部門があるように、データドリブンなビジネスを進めようと思う組織においては、データ品質を監視、評価、改善を組織的に推進することが必要になる。

4.4.4. データ活用の事例

データ活用の基盤を整備し、データ活用を行い、ビジネスへ寄与を行っている事例を紹介する。

(1)運輸業 A 社

<背景とデータ活用基盤の概要>

従来は、それぞれ単体の IT システムで運用を行っていた。IT システム毎に持つデータは以前からも利用していたものの、それらのデータを統合して捉え、顧客に対して、適切なタイミングで、適切な情報を提供しないと顧客体験は向上しないという議論が社内であり、当データ活用基盤が生まれた。

当基盤は、現場のメンバーが使う IT システムと基幹システムとが連携している。それぞれの基幹システムのデータは随時更新されるが、現場のメンバーがリアルタイムにデータを検索できる環境となっている。

リアルタイムに情報を参照できる基盤とは別に、データ分析の基盤は別に構築している。分析基盤へのデータ連携に関しては、1 日 1 回、夜間バッチで、データ項目を絞った上で、スナップショットを取っている。これは社内で顧客のデータ分析をする場合は、リアルタイム性を求められないためである。

<得られた効果>

①業務改善につながる情報発信

データ活用基盤にある情報を使って、適切な情報を提供できると、顧客の行動が変わるのではないかと想定し、いくつかの取り組みを行っている。

その中の 1 つとして、予約率が閾値を超えている場合に、それらを予約している顧客に対し、前日に事前の準備を促すための情報配信を行う取り組みを始めた。こうしたことを行った結果、情報配信した対象としていない対象で明確な差が出た。前もって情報発信をしておくことで、事前の準備に協力される顧客が増え、顧客の流れがスムーズになることで時間的な余裕が生まれた。結果、業務の改善にもつながることとなった。

このように、様々な情報を使い、適切なコミュニケーションを取ることをデータ活用しながら進めている。

②顧客へのタイムリーな情報提供

また、特定の業務については、リアルタイムが武器となる。それぞれの顧客に合った情報を適切なタイミングで届ける。例えば顧客に予約の利用準備ができたタイミングで、次に必要な情報を伝える、待ち時間の間にその場で利用できるクーポンを配布するという施策も行っているが、これは 10 分後に届いても顧客にとっては全く意味のない情報になってしまう。こうした業務では、即時性を重視しており、データを全体で統合していることで、実現できている。

(2)金融業 B 社

<背景とデータ活用基盤の概要>

データ活用はグループ全体での成長戦略の柱であり、グループ各事業で、今まで十分生かせなかったデータを一層活用することで新しい価値を生みだし、グループの利益拡大だけでなく、社会課題解決への貢献もしていきたいという理念も背景としてあった。

データ活用基盤としては、2種類あり、1つは、解決したい経営レベルの具体的課題の解決を目的として、関係するデータを収集し、その課題解決のためにカスタマイズした基盤となる。

もう1つは、グループ内のデータを統合的に収集/管理/活用する共通基盤として構築している。データが散在しており、全体感を把握しにくいという課題への対策として、グループ内で数年かけて取り組み、各部門の身近なデータ活用に関する課題を解決する環境として浸透しつつある。

総量としてどれくらいになるのか定まらず、またデータの要件や活用の仕方も変わっていくので、臨機応変に対応していく必要がある。ビジネス課題に柔軟に対応できるよう、いずれの基盤もクラウド環境に新規構築している。

<得られた効果>

①グループ内の連携による、顧客に寄り添った提案

既存事業だけに限ることなく、グループ各社が必要とする顧客に寄り添ったサービス実現を目指している。

グループ会社間をまたがる事例として、いくつかデータ連携に取り組んでおり、例えば、ある事業の顧客に対して、グループ内の他の商品・サービスを案内している。実施する上では、利用規約の範囲内でデータを連携・分析し、適切な情報提供や商品・サービスでの顧客アプローチを行っている。

②新しい示唆の検討

新たに取得している動的なデータやビッグデータを解析する基盤を活用し、新しい示唆を見出して顧客への有益な情報発信を行っていくことを検討している。

また、グループ各社でお互いが持っているデータに対する知見を補完しあい、関連する企業との共同研究、共同分析の取り組みも進めている。

4. 5. API

API とは「Application Programming Interface」の略であり、アプリケーションなどの一部を公開する手段を指している。インタフェースという特性上、ただ単純にアプリケーション全体を公開するというのではなく、「通信などの手段をもって機能呼び出せる」ようにするのが通常である。このように、開発したソフトウェアや機能をより容易に共有できるように連携できるインタフェースが次々と生まれるようになった。つまり API は、もともと異なるプログラムの間のインタフェースをつかさどり、プログラム間の通信をしながら、さまざまな環境下で作られてきたことになる。しかし、本節では、これらのように OS やミドルウェアが提供していたり、独自の特定ビジネスアプリケーション間の連携のために実装されたりしたインタフェースは範疇外とし、あくまで昨今の Web サービスを中心としたインタフェースを対象として話を進めることとする。

4. 5. 1. API の概要

本節冒頭で述べた通り、現在では技術の標準化や普及の背景から、連携を必要とするソフトウェアやプログラムの多くが HTTP プロトコル上などに Web サービスとして実装されることが多い。ここでは API を利用した開発環境や言語、細かな実装などには言及しないが、Web サービスを中心としたインタフェースを前提としながら API によって実現できることや、気を付けるべきことなどについて記述を進めていくことにする。

4. 5. 2. API 技術利用の背景

現在では社会におけるあらゆる機能が IT システムとして実装されており、またその IT システム間も連携しながら機能するように考慮されるようになってきている。DX 時代においてはその連携が API を用いた設計・実装を求めるようになってきている。その要因は主に2つあると考えられる

一つ目は異なる組織や企業間はもちろん、同じ組織や企業内の IT システムであってもそれぞれの IT システムに包含される様々な機能は互いに疎な関係での実装が求められていることである。様々な機能が独立して実装・運用されることにより、同じ機能の多重開発を避けることができたり、必要な時にその機能だけを呼び出して利用することができたりすることで、開発から運用までの効率を上げることができる。それだけでなく、市場やユーザーニーズの変化に伴う、一部機能の変更や追加もスピーディな対応がしやすくなる。

二つ目は組織や企業内で閉じないような連携が求められていることである。DX 時代においては単独の組織や企業が他と相互に連携することなく単独で激しい競争を勝ち抜いていくことは非常に困難となってきた。複数の組織や企業が互いの強みを生かしながら協力し連携しあって新たな市場を形成したり、イノベーションを起こしたりして成長をしていけるのである。そのためにはデジタル技術の面での連携は非常に重要で、それぞれが持ちうる IT システムの機能を互い

に利用しあえる状態にしておかねばならないのである。

このように現在の社会を支える IT システムの機能間連携やビジネス連携を考える上では API はとても重要な役割を果たしているといえる。

例えばあなたのオフィスで棚が必要となった場合を考えてみることにしよう。市販の出来合いの棚を買ってくるというのが簡単かもしれないが、棚を置く場所やそのサイズ・形状、使用目的など要件や要求が厳しければ、ニーズに合致した棚を手に入れるのは難しくなる。逆にすべて手作りをすると、設計から加工・組み立てまで道具をはじめノウハウも労力も必要となるだろう。ただ、その時に、木材のカットや加工など外部のサービスを使うことができれば要件や要求に見合った棚を安価に簡単に組み立てることができるかもしれない。

API はこれに似ている。業務の推進に必要なデジタル処理を、市販のパッケージやサービスを利用することもできるが、要件や要求が厳しければそれにちょうど見合ったパッケージやサービスを見つけることは困難であり、自ら処理を自営しなければならないだろう。ところが様々な外部の API や自ら用意した API などをも併せて活用できれば、よりスマートな構築が可能なのである。

4.5.3. API の活用により実現できること

API の活用により期待できることとはどんなことだろうか。すでに前項で述べた通り互いに疎な関係で機能ごとの実装が可能となることが挙げられるだろう。ここではこうした実装によって実現できることや享受できるメリットについて述べていくことにする。

<利便性>

API を利用して設計・構築されたシステムやサービスの一番大きなメリットは、自らのビジネスの利便性だけでなく、結果的に自らが提供するサービスを利用する側の利便性をも格段に向上させることができることにある。例えばある企業が提供するサービスに、オープン化されたユーザー認証サービスを採用することで、ユーザーは共通の認証機能を利用することで、サービスを利用するために新たに増加するはずだった ID 管理から解放されるようになる。

<効率性>

API を上手に利用することができれば、主に開発や運用面で効率性が格段に向上する。まず、複数の組織・企業のシステムで同一の機能の多重開発を避けることができるからである。サービスについても同様で、すべてを自営する必要がなくなり既存のサービスを利用しながら、独自性のある部分に特化した開発が可能となり、自らの強みの部分に投資を集中させることも出来る。

むろん自らも API を公開することで、他組織に対して（必要なら適正な対価と引き換えに）機能を利用させることで効率性を享受させることも可能である。他に同等の機能を有する組織や企業が無ければ、社会の効率性に役立つだけでなく競争優位にもたつことができるだろう。

<データ活用>

自らが持ちうるデータを囲い込むばかりでは、データの持つ本来の価値を有効に利用しきれているとは言えない。無論データの保全・保護・倫理上の配慮などは欠かすことはできないが、データを正しく処理・加工しながら正当に利用できる範囲を限りながら利用させることは社会の発展に役立つものである。API を介すことで、自らが持つデータを他に利用させたり、自らが収集蓄積することが困難だが他が所有するデータを利用させてもらったりといった、より広範囲でのデータの流通や活用が可能となる。

こうしたデータ群を組み合わせることで、新たな法則性を見出すことができたり、いままでなかったような新たなサービスを提供したりすることもできるだろう。このように様々な企業や組織がそれぞれ互いに疎な関係にある機能を実装し公開することで、それらを相互に利用しながら、より便利なシステムやサービスを提供することが可能になる。現在ではこのように API を通して、それを利用する第三者を含め相互に発展していく経済圏を「API エコノミー」と表現するようになってきている。

<セキュリティ>

API の利便性とセキュリティは、次項でも考慮点として記載していくがトレードオフ関係にあるという面は、確かにある。しかし、正しく企画・設計・構築することで、API によりそれぞれ独立した機能で構成されたシステムに、より強固なセキュリティを実現することもできる。

このように Web サービスを中心とした API は、組織や企業内の様々なシステム間での連携を柔軟に実装できるようにできる可能性があることがお分かりいただけるだろう。しかも、API をサービスとして外部に公開することで、組織や企業を超えた連携が期待できることになる。Web サービスとして実装された API の真の効果とは、組織や企業の内外の区別なくその機能を柔軟に連携できるところに現れるといえる。

4.5.4. API の活用時の考慮点

本項では API を活用していく上で注意しておくべき点について記載していく。API はあらかじめ配慮すべき点に留意をしながら活用すれば大きな効果が期待できるが、安易に用いればリスクにもなりかねないからである。API の特性上、範囲の大小はあっても公開されることが前提となるため、その制御や安全性の担保も重要なファクターとなる。検討段階から公開の対象や範囲、それに呼応した利用者の認証や機能利用に関する認可などは厳密に要件定義をしておくのがよい。

<設計>

API において設計は非常に重要な意味合いを持つ。企画・設計段階から API 自体の緻密な設計と徹底した統制、またそうした環境を実現・維持できる手段の検討を行っておくべきである。なぜならば API は一度定義し利用や公開を開始してしまうと簡単に改変することが難しくなるから

である。

命名規約の制定と順守は確実に行っておきたい。個々の API 自体の名称はもちろんパラメータ自体の名称まで規約に沿って進めることで運用や管理面での破たんを防ぐことができる。また個々の API にとどまらず API 群に対しても標準化設計の注意を怠るべきでない。API 群の管理を行っていく上で、常に同じ機能もしくは似たような機能の API をいくつも作らない・作られないように留意しておくべきである。API に定義の明確化と同時に将来にわたっての拡張性も意識し互いに疎な関係となるよう意識しておくといよい。

<性能>

性能面では、特にレスポンスやスループットを中心に定義を明確化しておくといよい。定義を明確化し標準化しておくことで、提供側・利用側ともに負荷や利便性の前提を明らかにしやすくなる。また負荷に対応するためどういった制限を設けるべきなのか、拡張や増強を考えるトリガーをどこに設けるのか、他の関連する API とのバランスどう考えるのかなど、定義がなければ具体的な設計や運用が困難となる。

<セキュリティ>

API は公開される前提で実装されるものであるため、事前に十分なセキュリティへの配慮を行わなければならない。仮に公開先が閉域網内であったり API の存在を知る組織内のみであったりしても、この考慮を怠ることはすべきではない。ましてや広く公開する API では不正な利用や攻撃への対策はネットワークでの防御などと組み合わせるなどして十分に行っておきたい。

API 側でも様々な技術的対策が必要となるが、ここでは API の利用を限定する例を 3 つに整理して挙げておく。

一つ目はアクセス権自体をコントロールする手段である。API を広く誰にでも公開するというのではなく、認証によって API へのアクセスを制御することも必要である。適切な権限をもった正しいユーザーにのみ公開することができるようになる。

二つ目はアクセス量をコントロールする手段である。特定のユーザーだけが API に対して膨大な負荷をかけ集中したアクセスを行わせないように制御する必要もあるだろう。またこうした手段は不正な API の探索などをブロックする目的でも使われている。

三つ目はアクセス範囲をコントロールする手段である。一つ目の例に似ているが、認証と認可によって、その API を通じて何にどこまでアクセスできるのかをきめ細かく制御する方法である。こうした制御はデータなどが事前に想定された適切な範囲にのみ流通することに役立っている。

リスクを前提としたセキュリティについて記載してきたが、認証や認可の仕組みが整備されていれば、API を通じて様々な機能、あらゆる組織が互いに信頼関係を結びながら連携することができる。この効果が絶大であることは、API を背景としたさまざまなエコシステムがつぎつぎと生まれていることが証明していると言える。API の戦略がビジネス戦略そのものとなるわけである。

<運用・維持・メンテナンス>

冒頭でも示した通り API を提供する側としては、API のサービスをいったん公開し使われ始め

ると、安易に停止したり変更したりすることが難しくなる。特に広く公開された API はつねに組織外からもアクセスが発生し自己の管理下だけでは簡単にコントロールできなくなるため慎重に企画・設計・実装することが重要となる。さらにサービス利用者へ及ぼす影響も考えて運用をしていかなければならない。こうした非機能要求である信頼性についてもあらかじめ広範に考慮を進めておくのが良い。

API サービス提供者側だけでなく API のサービスを利用する側も、サービス提供者側の運用・維持・メンテナンスに関心を持ち十分な調査を行った上で、利用上の要求を満たせるかどうかを確認しておく必要がある。通常の利用だけでなく、万が一の場合の障害切り分け、利用できない場合の措置なども考慮しておく必要がある。また利用する API サービスが終息したり、自らのソフトウェアの改修を伴う API 改変などがあつたりした場合のリスクの範囲やリスクヘッジ方法まで検討しておくといよい。

<API ゲートウェイの提供や利用>

API ゲートウェイとは、複数の API の管理や実行をより容易に実現をするためのサービスである。API ゲートウェイは API 利用者と様々な API サービス群の間に介在し、API 利用者から受け取った依頼(リクエスト)を、様々な API サービスに振り分ける役割を果たす。

現在では多くの IT システムが複数の API を用いて構成されている。IT システムの多くは、例えばユーザーを識別する「ユーザー認証」、ユーザーのロールやそれに応じたポリシーをつかさどる「認可」、個々のアクセスの負荷などを考慮した「レート制限」、サービスや機能の利用に応じた「課金」などの API が外在していると考えられる。通常は利用する API ごとにそれぞれ個別に呼び出したり通信を制御したりといった管理を行う必要がある。

API ゲートウェイを利用した場合、API 利用者は主として API 群をまとめて管理する API ゲートウェイとだけ通信を行えばよくなる。つまり API ゲートウェイを使った場合は、それぞれの API サービスの呼び出し管理や通信を任せることができることになり、IT システム開発の複雑な処理を単純化するのに役立つわけである。API ゲートウェイの技術は現在も発展途中にあるが、今後は利用側にとって複数の API の管理機能の役割のみならず、API サービスの追加や廃止の管理といった面で利便性が向上していくと考えられている。同様に提供側にとっても、API 群へのアクセス統計や分析、API 利用がセキュリティポリシーに則っているか、違反があればアラートをあげるといった監視機能なども徐々に充実し、さらに API 技術を発展させていくものと考えられている。

<API とビジネス戦略>

いざ API を公開するとなると、本来自らのみが利用できていたものを他者が利用できることになりビジネス上の優位性を維持できなくなるのではという危惧もあるだろう。これまでのビジネス常識ではそうした判断もされてきて当然である。しかし DX 時代においては、逆に、API を利用して市場を囲い込むことなども、大きな戦略上の選択肢となってきた。

昨今目立つのは API を公開することで利用者を増やし先んじて市場を形成してしまうケースである。普及率が上がればビジネス面では先行有利にはたらくことになる。さらに最初は無償で市場占有率を上げて置き、のちに有償サービス化するケースも多く出てきている。

4.5.5. スサノオ・フレームワークとの関連

この項では第4章冒頭で述べた「スサノオ・フレームワーク」とAPIとの関連に触れていく。スサノオ・フレームワーク上でAPIと表記されているのは【F】の部分のみであるが、このAPIは単一のAPIではなく、API群を示している。このため図4.5.1ではAPI（群）を通して他のあらゆる要素と結ばれており、相互に関わりを持っている表現となっている。

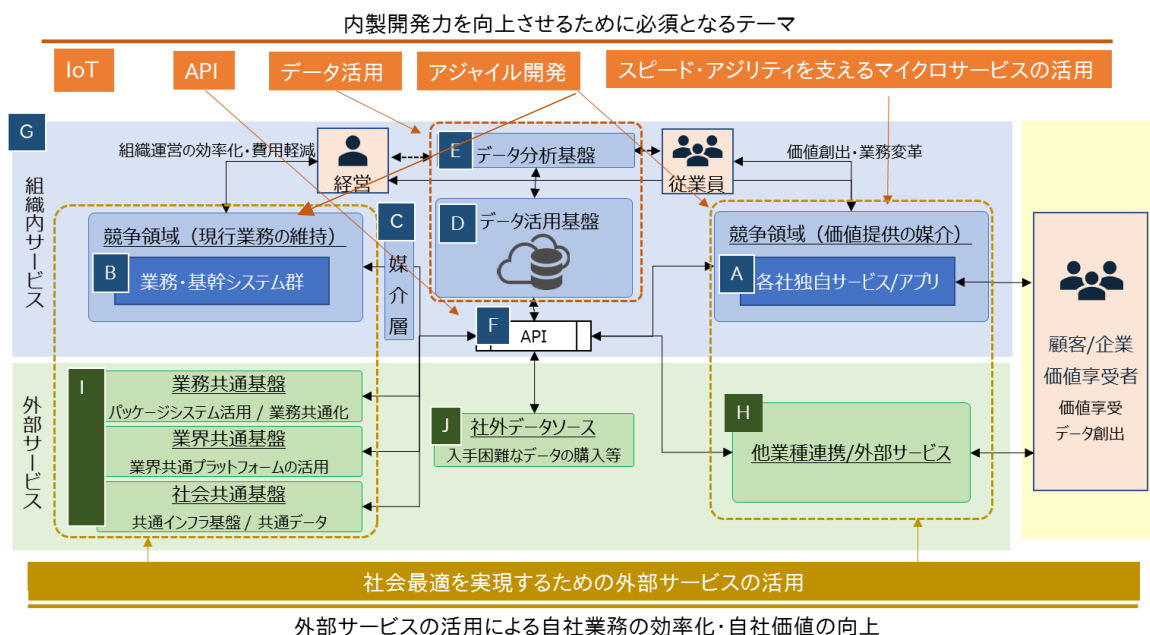


図 4.5.1 あるべき IT システムを実現する技術要素群「スサノオ・フレームワーク」（再掲）

例えば競争領域の独自アプリケーション(A)と外部サービス(H)は、API を利用することで自社が競争領域に、協調領域の機能を取り入れるための連携を図れることを示している。例えば、外部サービス(H)で広く一般利用されているユーザー情報を API 連携し、競争領域の独自アプリケーション(A)のログイン認証などに利用することで、自社でユーザー情報を持つ必要がなくなり、ログイン認証機能の開発量の削減や情報漏洩のリスクの低減が可能となる。いずれも新たな価値を市場に提供していく機能であるが、外部のサービスも利用しつつ自らの持つ強みを生かしながらよりアジリティを持ったサービスが可能となる。多くの人が利用している通信サービスなどと連携する API などを用意することで、例えば競争領域の独自アプリケーション(A)もより広範囲に利用できるようになるわけである。すでに世の中には、例えば高性能な分析処理などに代表される便利で優れた機能が数多く実装されているものもある。

また実装までのリードタイムの短縮やサービス化の加速や機能拡張なども期待できるだろう。このような利用方法ではどのようなサービスを実現したいのか、どんな機能が必要なのかといった要件を明確にし、外部のサービスの中から要件を満たせるものを選定しなければならない。ま

た選定したサービスと連携できるよう自らのサービスも API を調整していくことが重要となる。

ほかにも競争領域と協調領域の連携のパターンとして、現行システム(B)と外部共通基盤(I)の API 連携がある。現行システム(B)は主に組織内業務を実現するシステム群であるが外部サービスなどを活用し、より柔軟で安価に維持することも可能となる。基幹業務のうち協調領域にある業務はできるかぎり標準化しすでに世の中に存在・実装されているサービスに移行することで IT システムの維持体制・コスト・リスクを低減することが期待できる。例えば外部共通基盤(I)として業界標準の EDI(電子データ交換 : Electronic Data Interchange 標準化された規約にもとづいて電子化されたビジネス文書を企業間でやり取りすること)などを利用することで現行システム(B)の受発注機能を標準化するという方法がある。ただし一部の IT システムを外部サービスに移行するだけでは連携の効果を果たさない。IT システムだけではなく実業務自体の切り分けや整理・標準化がどこまで徹底できるかがポイントとなる。

もう少し複雑な例としてデータ活用基盤(D)、データ分析基盤(E)、社外データソース(J)の関係を挙げてみることにしよう。これまで多くの組織では事業などに有益なデータをデータ活用基盤(D)により多く蓄積してきただろう。またこうしたデータについてデータ分析基盤(E)を用いてデータドリブンな判断ができるように配慮してきた組織も多い。しかしながら自らの組織で収集できるデータは限られている。社外などのデータソースから分析に必要なデータを購入するなどしてハイブリッドな活用をする段階でもその連携に API が有効に働くことになる。

こうしたスサノオ・フレームワークのような思想は新たな戦略やシステムにだけ適用できるわけではない。例えば通常なら変化の激しい領域にある競争領域の独自アプリケーション(A)と、比較的变化がゆっくりである現行システム(B)など、通常なら足並みがそろいにくいシステム間であっても API を用いることでデータを流通させたり、機能を活用したりすることが可能となるわけだ。図を見ていただくと、競争領域の独自アプリケーション(A)と、現行システム(B)の間に媒介層(C)が存在しているのに気づくだろう。ビジネス上の都合から、ある日突然各社独自サービス／アプリを利用しようと思っても、既存の業務・基幹システム群を IT システムもろとも一掃したり全般にわたって改変したりするというのは困難である。なぜなら多くの場合、IT システムの複数の箇所に対して再設計や開発・再構築が必要となったり、それによる他システムへの影響を考慮しなければならなかったり、膨大なテストを実施しなければならなくなるからである。しかし既存の業務・基幹システム群に大きな変更を行わず、API を内部もしくは外部に追加することでデータ流通の自由度を向上させたり、既存の IT システムの機能を利用・活用したりする事ができるようになる。これを実現するのが API 媒介層の役割である。さらに図をよく見ると外部共通基盤(I)、外部サービス(H)といった領域にある外部サービスとも API 媒介層を介して連携ができることがわかるだろう。

ただし、ただ API が用意されていれば（もしくは用意された API を活用すれば）よいという事ではない。これまで記載してきたとおり、API の活用は関係する IT システムとどの様に連携していくか戦略をたてて行う必要がある。

4.5.6. API 全体管理とスサノオ・フレームワーク領域別の特徴

この項では API を活用している企業の事例を元に「API 全体管理」と「スサノオ・フレームワーク領域別の特徴」について説明する。「API 全体管理」では API を活用・検討する上で、技術的な観点と組織的な観点それぞれについて、「スサノオ・フレームワーク領域別の特徴」では IT システムにおける API はどのような特性を持っており、どう活用され、どうあるべきかスサノオ・フレームワーク領域別について説明する。

■API 全体管理

API 全体管理は競争領域の独自アプリケーション(A)などで少数の API を部分的に活用しているだけであれば必要とならない場合もある。しかし競争領域の独自アプリケーション(A)などで多数の API を活用する段階や、様々なスサノオ・フレームワーク領域で活用する段階になる場合、もしくは今後多数の API を活用する見込みがある場合など、多くの API を個別に管理していくことはとても煩雑となる。場合によっては管理が行き届かず事実上管理できていない状況となり、利用可能な API が埋没してしまい API が再利用できないといったケースや一部の API にセキュリティ対策が漏れるといったケースなどが発生する。そのような状況になると、API を再利用することで開発スピードを向上することや、API の非機能要件を制御して安定した IT システム運用を行うことなどができなくなる。そのため、API を全体管理する必要が出てくる。API の全体管理は、技術的な観点と組織的な観点で考えることができる。

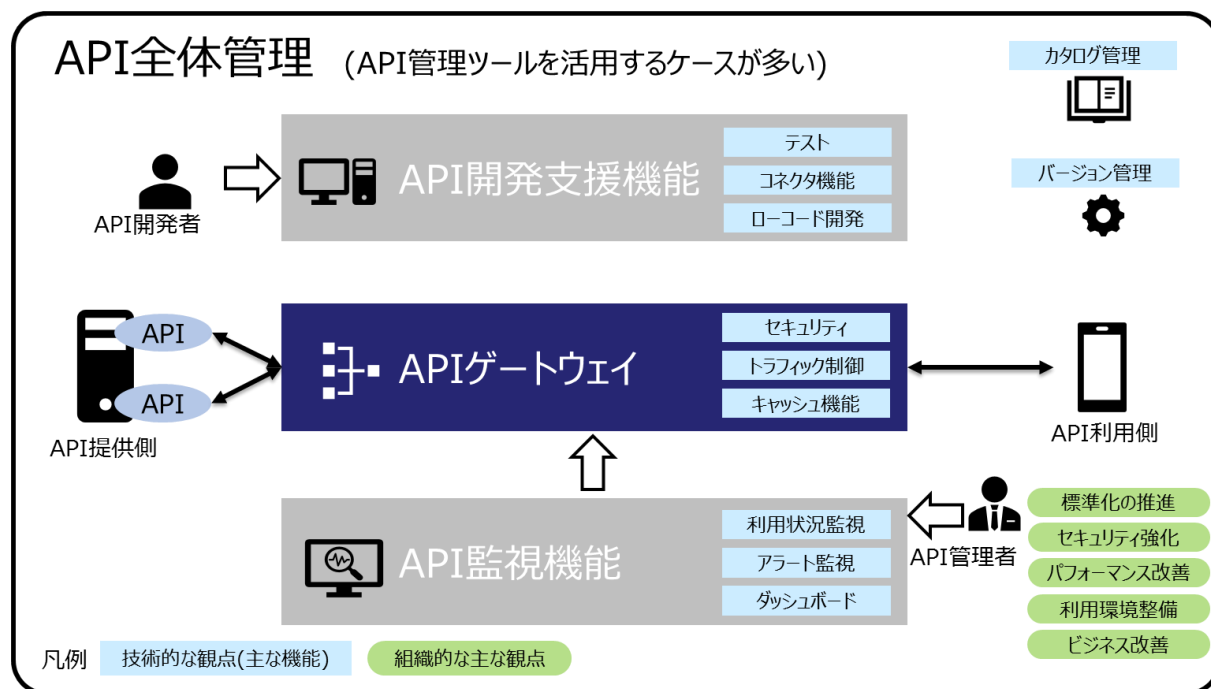


図 4.5.2 API 全体管理

(1) 技術的な観点

技術的な観点では、API 管理ツールなどで API ゲートウェイ (API 利用側と API 提供側の

間に位置し、API 群の管理や実行時の振る舞いを容易にする仕組み。※4.5.4 項< API ゲートウェイの提供や利用>参照) を構築して API 全体管理を行うことが一般的である。

ここでは「API 管理ツール」は「API ゲートウェイ」に加えて、API を少ないコード(ローコード)でプログラム実施を可能にする機能や、画面が無くても API のテストを可能にする機能などで開発を支援する「API 開発支援機能」、API の動作を監視する「API 監視機能」などを含んだものと位置付ける。(API 管理ツールによっては API 開発支援機能や API 監視機能がないものもある。また、API 開発支援ツール、API 監視ツールとして単体で提供されるものもある)

<API ゲートウェイ>

API ゲートウェイは API 提供元の API を利用する際の共通の入口となる機能以外にも様々な機能があるが、API に共通的に実施することが必要な設定を一元管理し、それによって API の利用を簡素化、標準化することなどのために利用されることが多々ある。例えば、セキュリティの観点から「認証・認可」やアクセス権・アクセス範囲を設定する「アクセス制御」を活用することで、利用する側も機能ごとに、サービスを提供する側も同等レベルで安全なアクセスを実現させることが可能となる。

最も簡便な「アクセス制御」の例としては、API アクセスキーを用いる方法がある。API 提供側は API を通常の呼び出しでは利用できない状態としておき、利用させたい API 利用者のみ、予め発行した API アクセスキーをアクセス時に送信させ、これを API 提供側が確認することで利用者を限定した API 公開をするといったことが可能となる(図 4.5.3 参照)。

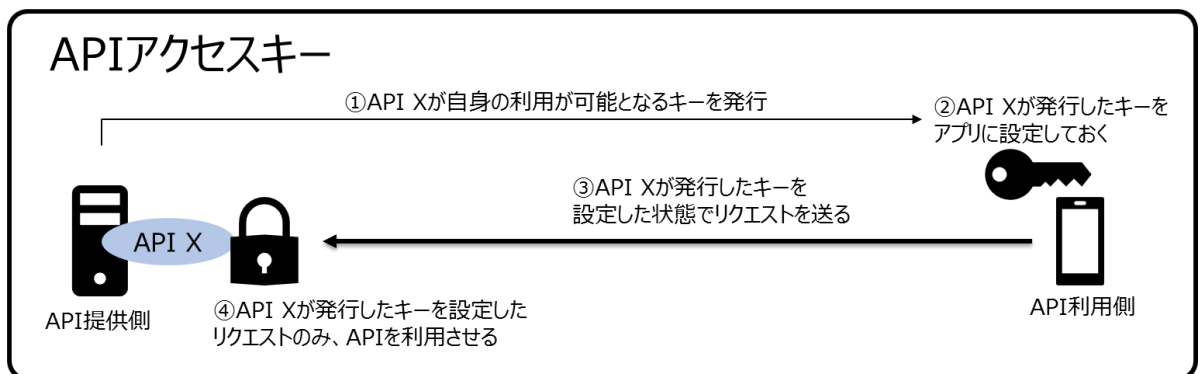


図 4.5.3 API アクセスキー

アクセス制御にはセキュリティ強化のためアクセストークンを活用する方法がある。API 利用側に API 提供側へアクセスを許可するためのアクセストークンというものを API 提供側とは別のアクセストークンを生成する認可サーバなどから、API 利用側に払い出す。API 利用側は、認可サーバから提供されたアクセストークンで API 提供側にアクセスする。

また API ゲートウェイではアクセス制御の他にも急激なアクセス増加の遮断や、API の呼び出し回数を制限する「トラフィック制御」といったコントロールができるものもある。

<API 開発支援機能>

API 管理ツールが提供する API 開発支援機能を利用することで、API 開発者の API における設計・開発時間を短縮することができる。例えば API を開発する上で必要となるデータソースからデータを取得する場合、目的のデータソースへの接続処理は API 管理ツールから提供されているコネクタ機能（図 4.5.4 参照）を活用することで実現可能となる。これにより設計・開発量を抑えることができる。

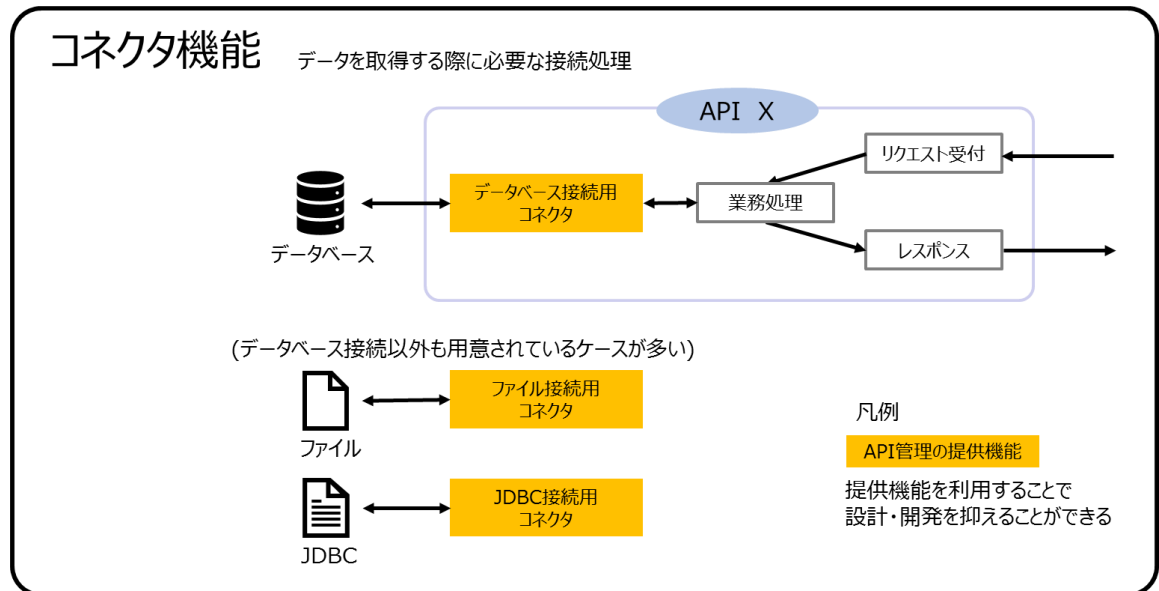


図 4.5.4 コネクタ機能

コネクタ機能を活用する場合など、API を利用しデータを取得する上でトランザクションの一貫性が求められる場合、1 つの機能の中で複数の API を利用し API がそれぞれテーブルにアクセスする設計だとトランザクションの一貫性の保証ができない可能性がある。そのため一つのトランザクションは一つの API で処理を行うといった方法もある。

また、データストアの読み取りと更新の操作を分離する CQRS（Command and Query Responsibility Segregation）などを実装していくことが重要である。リクエストは、リクエストで受け止めて MQ（メッセージキュー）に一旦いれて、セッションとして終わりにする。MQ に入ったキューがしっかりトランザクションになっていき、エラーが起きたらロールバックするという仕組みを実現するといった方法もある。単純なロールバックでは元の状態に戻せない場合は、トランザクションで実行した操作の逆向きの取り消し操作を行うといったことなどが必要となる。

<API 監視機能>

API 監視機能では API のトラフィック量、応答速度や API で発生したアラートなどが可視化できる。API は社内業務遂行や顧客へのサービス提供を行う上で、ミッションクリティカルな役割を担っているケースも多い。そのため API 監視機能で API の運用状況を一元的に把握できる状態にしておき、トラブルを未然に防ぐ対策を行うことが重要である。またトラブルが発生しても迅速な対応ができるようにしておくことが IT システムの運用

上求められる。

API ゲートウェイを活用し API を一元管理できれば、API の監視は API ゲートウェイだけを対象とすることが可能となる。

<その他機能>

その他にも API 管理ツールを利用することで API に関する情報を登録・参照できる API カタログ管理や API のバージョン管理を一元的に行うことなどが可能となる。

(2) 組織的な観点

組織的な観点では、企業内に API を全体的に管理する役割を持った組織(API 管理者)を作ることが望ましい。API 管理者の役割として API 設計・開発時には API を利用する IT システム担当者や部署の間に立ち、データの利用側と提供側の要件を確認・調整してパフォーマンス定義をする。

<標準化の推進>

API 設計の標準化を推進し、標準仕様に沿った設計が行われているかを確認することが API 全体管理をする上で重要となる。例えば、API 名称や API パラメータの命名規約を定め、標準化を推進することで、開発者は API を検索しやすくなる。そうすることで、API のメリットである API 再利用が促進されることで開発スピードを向上させることができる。

また、同様の目的を持つ API の乱立を排除するため、社内の API を目的別に把握し、既存の API の精査や同様の目的を持つ API の新規作成を抑止することが重要である。

<セキュリティ強化>

API 開発が完了し API をリリースする際はセキュリティを強化するため、API アクセスキー(図 4.5.3 参照)の発行や、API アクセス権及びアクセス範囲の設定を監督する役割を担うことも重要である。

<パフォーマンス改善>

API 運用時には、安定した IT システム運用を実現するために、API の運用監視やパフォーマンス改善を主導する役割となる。

例えば、API アクセス量が増える場合、API ゲートウェイを複数用意し、業務別に分散させるといったことを主導する。

<利用環境整備>

社内の API の利用側の視点に立ち、社内の API 開発者による API 情報を提供するサイトの運営や、API に関する情報を一元管理する API カタログ管理など API を利用しやすい、開発しやすい環境を構築する役割などが API 管理者の役割として必要となる。

<ビジネス改善>

技術の進化とともに API も日々進化を続けている。現状のビジネスをより良くするために最新の技術動向に目を配り、その最新の技術が自社のビジネス改善のために活用できないかを検討することも、API 管理者として重要な役割となる。

例えば次項 4.5.7 で紹介する(3)小売業 C 社事例のように非同期処理に API を活用するといった方法がある。(「非同期 API」と呼ぶこととする)非同期処理ではクライアントはシステム A に処理を依頼したらシステム A の処理結果を待たずに完了することができる。(図 4.5.5 ①参照) これによりクライアントの処理時間は大幅に削減可能となる場合がある。そして、システム A とシステム B の連携はシステム A の処理終了をシステム B が判断するために定期的に API を起動させ実現させる。(図 4.5.5②参照) これによりシステム A とシステム B を連携させるためにシステム A に改修を加える必要がなくなる。

また、画面の表示速度をあげることでユーザー体験(UX)を向上させたい場合に、画面表示に必要なデータを必要最小限だけ取得することで通信量を減らし負荷を下げる方法がある。API のクエリ言語である GraphQL などを活用することで全てのデータを取得し負荷が高くなることを避け、必要最小限のデータのみを取得することが可能となる。

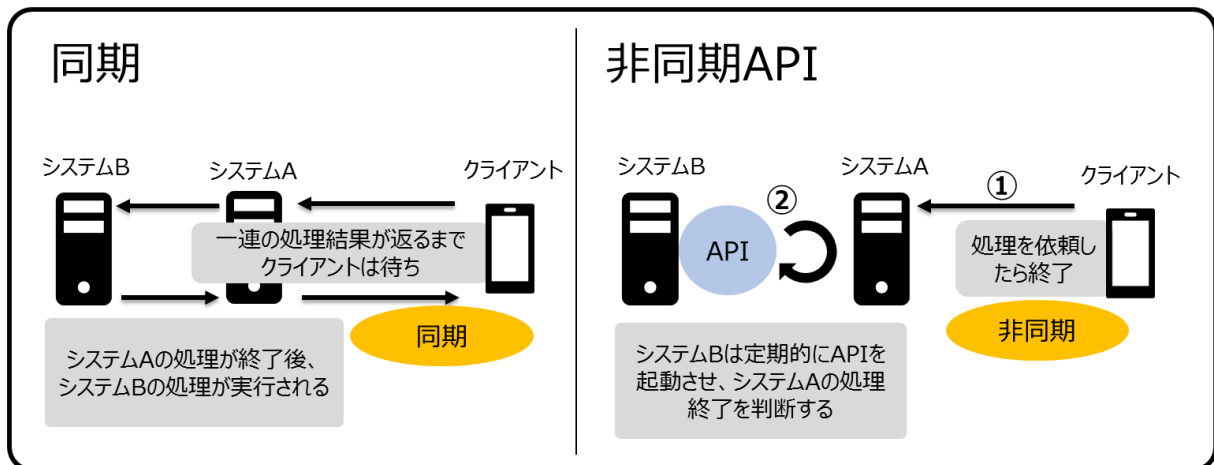


図 4.5.5 非同期 API

■スサノオ・フレームワーク領域別の特徴

API 全体管理の項でも述べたように少数の API を活用する段階から、統合的に多数の API を活用して DX を推進する段階になると、API 全体管理を行い、API を活用しているスサノオ・フレームワーク領域に関わらず、API に関する管理ツールの導入や、プロセスの一元化・標準化を進めていくことになる。そのため、API 活用に関する取り組みや工夫を、全てのスサノオ・フレームワーク領域に、基本的には一律に適用することになるが、スサノオ・フレームワーク領域ごとの特徴や特性も考慮する必要がある。以下に、スサノオ・フレームワーク領域ごとに考慮する点をまとめた。

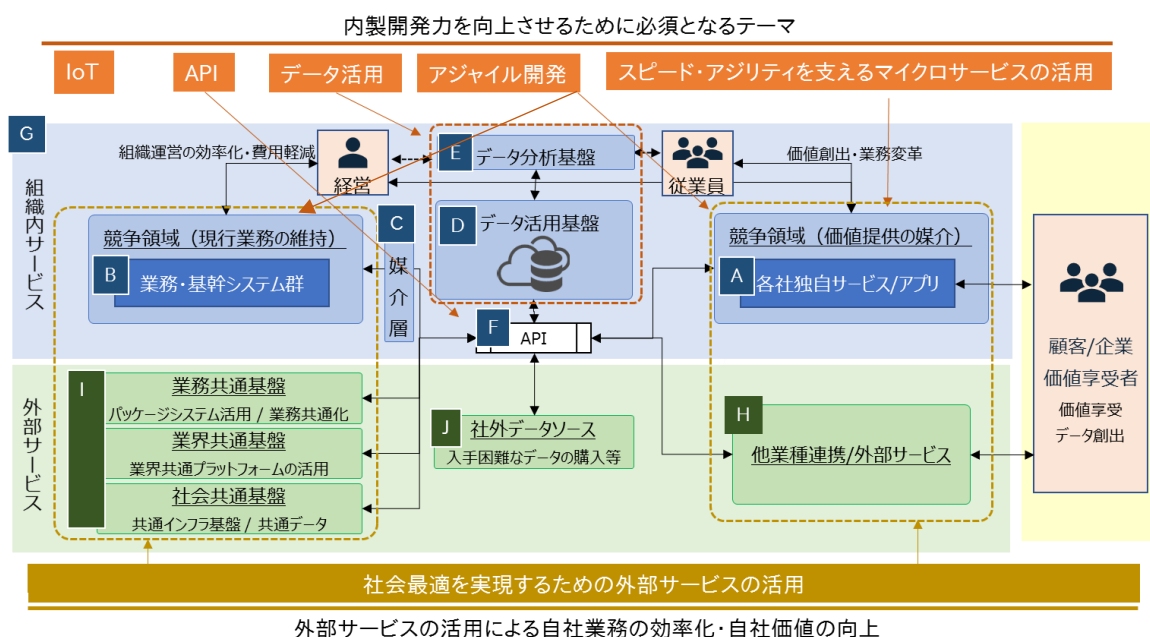


図 4.5.6 あるべき IT システムを実現する技術要素群「スサノオ・フレームワーク」(再掲)

(1) 競争領域の独自アプリケーション(A)

競争領域の独自アプリケーション(A)は、顧客向けにサービスを提供する領域である。この領域は、現行システム(B)と比較すると、歴史が浅く新たなシステムや機能の開発が行われているケースが多い。そのため、新規に API を開発して活用するケースが多い。また、比較的变化の少ない現行システム(B)に比べ、顧客向けのサービスを向上させるために、激しい変化に柔軟に対応していくことが求められる。

競争領域の独自アプリケーション(A)では、将来的に様々なビジネスにも対応できる構成にしておく API ファーストの考え方が重要である。例えば、EC サイトで商品管理や在庫管理などを担うバックエンド部分(現行システム(B))と、顧客が照会・操作するフロントエンド部分(競争領域の独自アプリケーション(A))をあらかじめ分離して開発・運用し、バックエンドとフロントエンドを API で連携する「ヘッドレスコマース」に代表される API 連携を前提とする設計思想を採用することも有効である。それにより、フロントエンド部分(競争領域の独自アプリケーション(A))を新規作成もしくは改修する際に、バックエンド部分(現行システム(B))の改修が少なくなる(もしくは無くなる)ことにより、開発スピードを向上できる。

(2) 現行システム(B)

現行システム(B)は既存で運用している IT システムが多く、API を利用していないケースもある。API を利用している場合であっても API の標準化を考慮せずに利用しているケースが多い。例えば、競争領域の独自アプリケーション(A)で顧客向けサービスを提供するために、現行システム(B)のデータを活用することが必要になる場合、現行システム(B)の以前から利用している標準化の考慮がされていない API をそのまま利用し、競争領

域の独自アプリケーション(A)とデータ連携するケースである。競争領域の独自アプリケーション(A)で必要になるデータが、現行システム(B)の様々な IT システムに点在している場合、現行システム(B)の個別の API ごとにデータ連携を構築すると、開発効率が悪くなる。また、現行システム(B)は、様々な IT システムや業務で利用されているため、変化に柔軟に対応することが難しいケースが多く、この構築に現行システム (B) の改修が伴うことになれば問題はさらに膨らむ。そのため、現行システム(B)の API は既存のまま利用し、現行システム(B)と競争領域の独自アプリケーション(A)の間に API ゲートウェイを構築(この API ゲートウェイを構築した基盤を「API 基盤」とする)し、API 接続を標準化する事例が増えている。そうすることで、現行システム(B)と競争領域の独自アプリケーション(A)のデータ連携を構築する際、現行システム(B)の個別の API ごとにデータ連携するのではなく、競争領域の独自アプリケーション(A)は API 基盤上の標準化された API と連携すれば良くなり、開発効率を向上させることが可能となる。また、パフォーマンスやセキュリティなどの非機能要件の制御を一元管理することや現行システム(B)を変更せずに競争領域の独自アプリケーション(A)の開発要件を吸収することができるなど様々なメリットがある。

新たに構築した API 基盤に、既存の API 接続も集約していくことになるが、すべての既存の API を一気に標準化してしまうと、従来の API 仕様に慣れている現行システム(B)の開発者の中には、新たな仕様を習得するための時間がかかるケースがある。そのため、段階的に API を集約していくことや、従来の API 仕様を踏襲するなど柔軟に対応することも場合によっては必要である。そうすることで、API 基盤構築・運用時の混乱をコントロールすることが可能になる。

(3)データ活用基盤(D)、データ分析基盤(E)

データ活用基盤(D)では社内の様々な IT システムやデータソースからデータを集約し、データ分析基盤(E)ではビジネス部門が自分たちで分析できるようにするため、データ分析ツールにデータを供給する役割を API が担うこともある。API がデータベースに直接アクセスしてデータを抽出する形になることも多いため、API 管理ツールなどが提供するコネクタ機能(データベースなどへの接続に関する複雑さを軽減する仕組み)を活用して、データベースとの接続作業を容易にすることが可能となる。

(4) 外部サービス(H)

外部サービス(H)では社外サービスプロバイダが提供する API を利用することになる。そのため非機能要件を社外サービスプロバイダに依存する部分が大きくなる。しかし、社外サービスプロバイダにただ依存するだけでなく API 監視ツールを活用して、外部サービス API のパフォーマンスなどを監視し問題を発見できるようにしておくことが重要である。また問題の発生に備え対象の API と関係する IT システムを事前に洗い出し対処方

法を検討しておくことで、問題が発生した場合の初動を早めることができる。例えば問題が発生した場合に備え、代替サービスを用意しておくことやサービス緊急停止の案内を用意するなどを検討しておくことが望ましい。また、社外サービスプロバイダとの SLA

(Service-Level Agreement) 締結時にセキュリティ要件や API が利用できなくなった場合の対応を確認し、必要に応じて提示された性能上限を増やしてもらうなどの交渉をすることも必要である。

(5) 外部共通基盤(I)

外部共通基盤(I)では SaaS システムが提供する API を活用することが多い。社内で開発した API ではないので、SaaS システムが提供する API の中身がどのようなになっているか、詳細を把握することが難しい。そのため、SaaS システム提供 API を介して複雑なデータや大量のデータを連携する際、思ったようなパフォーマンスが出ない場合の対処方法を考えておく必要がある。自社でできることとしては、SaaS システムの性能上限を事前に把握し、性能上限を超えないようにするために、実施できる対策を考えることや、性能上限に近づいた場合の対処方法を検討しておくことなどがある。それでも自社として対処が難しい場合は、事前に SaaS システム提供ベンダー側に常時もしくは一時的に性能上限を開放する交渉をしておくことも場合によっては必要である。

(6) 社外データソース(J)

社外データソース(J)では社外のデータプロバイダーなどが提供する API を介して自社が保持していないデータを取得する。社外データを組織内サービス(競争領域の独自アプリケーション(A)、現行システム(B)、データ活用基盤(D)、データ分析基盤(E))に接続して取り込む際、組織内サービスの構造が複雑な場合や社内データの活用が進んでいない場合は社内データを連携し希望するデータを抽出することも難しい状態であるため、社外のデータを活用することが難しいケースがある。そのためまずは組織内サービスの整備や、社内のデータの活用を先行して行い、その後社外データの活用に取り組むといった方法もある。

また、社外から API を介して取得するデータがビジネス上ミッションクリティカルである場合は、外部サービス(H)の API と同様に、API の監視ツールで、外部サービス API のパフォーマンスなどを監視し問題が起こった場合の初動を早める準備をしておくことが重要である。

(7) 媒介層(C)

媒介層(C)は競争領域の独自アプリケーション(A)と、現行システム(B)の間に位置するが、現行システム(B)に大きな影響を与えずに再設計や開発・再構築を行う場合などに API を活用する。詳しくは「4.8.現行システムからあるべき姿への移行」を参照頂きたい。

(8) API(F)

スサノオ・フレームワークでは各領域のインタフェースとなるため、便宜上中央に API(F) を一つ配置しているが、実際は各領域それぞれに API は存在する。各領域の特徴については (1)～(7)までを参照頂きたい。

4. 5. 7. API 活用の事例

この項では API を活用している企業のアプローチや工夫に関する 4 つの事例を紹介する。各事例にはどのような役割、立場の方にとって参考となるものかを示すための<役割>や、事例の全体像を簡潔に表現した<概略>を設けている。事例を読み進める上で参考にさせていただきたい。

留意点として各事例は実際にスサノオ・フレームワークをモデルに構築しているわけではなく、便宜上スサノオ・フレームワークの各領域として例えている。

表 4.5.1 各事例で示す役割一覧

事例	役割	
	API管理者	API開発者
(1)情報・通信業A社	○	
(2)保険業B社	○	
(3)小売業C社	○	○
(4)製造業D社	○	○

(1)情報・通信業 A 社

<役割> API 管理者

<概略>

スサノオ・フレームワークの競争領域の独自アプリケーション(A)と現行システム(B)の連携を API 基盤 (API ゲートウェイ) に集約し、接続仕様の標準化推進により開発者の個々の業務の API 仕様や使い方を理解するための時間を短縮でき、競争領域の独自アプリケーション(A)の開発短サイクル化と安定したサービス提供を実現した。

<背景・課題・ニーズ>

主力事業で獲得した社内に保持している様々なデータを活用して、顧客への提供サービスを拡充し他事業を育成・拡大していく企業方針を推進していた。しかし、現行システム(B)上に様々な仕様の違う API が存在し、競争領域の独自アプリケーション(A)を開発する際は開発者が API 仕様を都度理解する必要があり、多大な時間がかかっていた。そのため、競争領域の独自アプリケーション(A)が現行システム(B)上の様々な仕様の違う API と個別

に接続する方式では時間がかかりすぎ、顧客への提供サービスをより早期にリリースする上で限界を感じていた。

また、何千万人単位の顧客が提供サービスを利用しており、サービス要求水準が高いため問題が発生した場合、大きなクレームになることから、安定した API 運用を実現する必要があった。

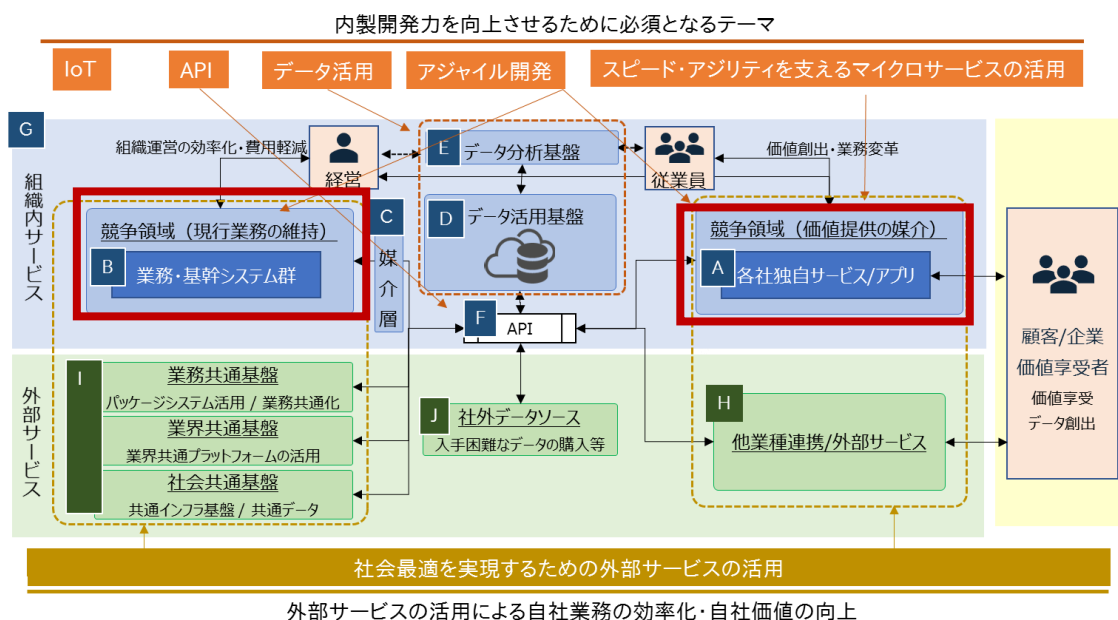


図 4.5.7 本事例の API を活用したスサノオ・フレームワーク領域

<取組みとその効果>

①短期開発の実現

競争領域の独自アプリケーション(A)と現行システム(B)の連携を、図 4.5.8 のように API 基盤 (API ゲートウェイ) に集約した。競争領域の独自アプリケーション(A)の開発者が、現行システム(B)の情報に基づく機能を開発したい場合、現行システム(B)の API を個別に呼び出すのではなく、API 基盤 (API ゲートウェイ) 上にある必要な機能を持つ API 群を呼び出せば良くなった。API 基盤上の API 群設計仕様は、標準化されており、現行システム(B)の個別 API との接続を行う必要がなく、仕様の違う個別 API を理解しなくて良いので、開発者が API の仕様を理解する時間を 70%以上削減することに成功した。その結果、競争領域の独自アプリケーション(A)の開発サイクルを短縮することができた。

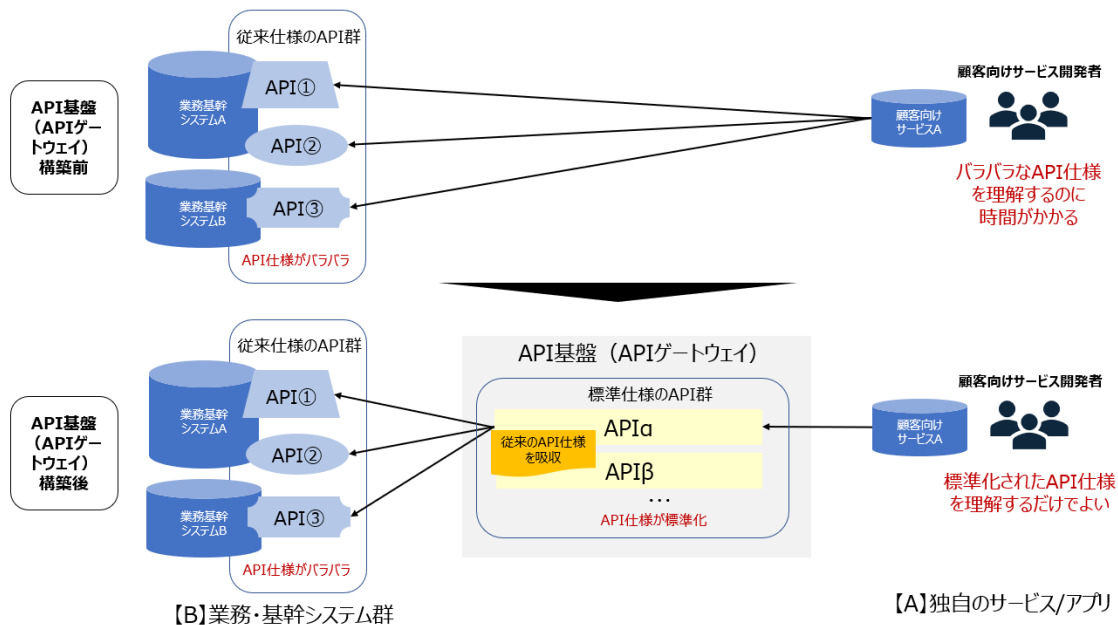


図 4.5.8 API 基盤構築前後の比較

また、API 管理者は短期開発の API の施策として API 開発者が自ら社内の API に関する情報を取得できるよう社内サイトを構築している。API 開発者が社内存在する API 仕様について自己解決できるようになり API 管理部門へ問い合わせる時間が 10 分の 1 程度となった。これにより API 開発者、API 管理部門の双方の作業効率化が実現できるようになった。

②安定したサービス提供

安定した API 運用を行うために、API 基盤に API 管理ツールや API 監視ツールを導入した。API 管理ツールでは API 負荷制限、API コネクタ機能（データベースなどへの接続に関するコードの複雑さを軽減する仕組み）の活用やセキュリティ設定などを行い、その中でも安定運用を実現するためにトラフィック量の制限を行っている。現行システム(B)は、急激なトラフィック増加には柔軟に対応できないため、競争領域の独自アプリケーション(A)側に最大アクセス量を出してもらい、現行システム(B)とトラフィック量を合意して API の設計を行った。

また、API 監視ツールでリアルタイムにトラフィック量、応答速度、エラー件数など API の稼働状況をダッシュボード上で監視・分析することで、IT システム障害を起こさず安定した API 運用を実現した。

(2)保険業 B 社

<役割> API 管理者

<概略>

スサノオ・フレームワークの幅広い領域で API 活用が増加する中、API ゲートウェイ構築によりガバナンスを利かせ一貫したパフォーマンスやセキュリティを確保しつつ、安定した IT システム運用、品質向上、データ活用促進を実現した。

<背景・課題・ニーズ>

金融・保険業として、以前から安定した IT システム運用を実現するために、特にパフォーマンス、可用性、セキュリティなど非機能要件に注意を払っていた。スサノオ・フレームワークの競争領域の独自アプリケーション(A)、現行システム(B)、データ活用基盤(D)、データ分析基盤(E)、外部サービス(H)、外部共通基盤(I)、社外データソース(J)の領域で、IT システム間連携に API を活用していたが、IT システム同士が API を介して直接接続する形であった。また、社内システムが一層増加しており一つの API が数十を超える IT システムに利用されることもあった。欧米では、API 規格を定めて標準化を行い、IT システムのパフォーマンスやセキュリティをコントロールするやり方が主流になってきたこともあり、今後 IT システム接続が社内・社外問わずさらに増加することを見据え、IT システム間のデータ連携状況を把握するためにも、API ゲートウェイを構築することにした。これにより IT システムのパフォーマンスのコントロールやセキュリティへの対応だけでなく、IT システム開発の生産性や運用の品質も向上すると考えた。

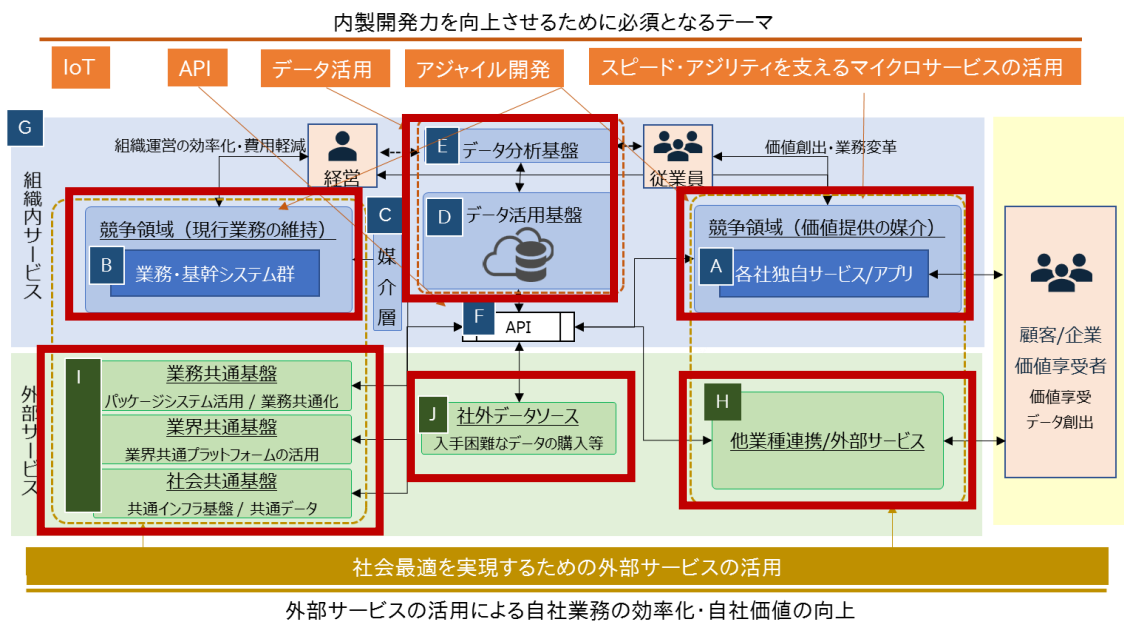


図 4.5.9 本事例の API を活用したスサノオ・フレームワーク領域

<取組みとその効果>

①安定した IT システム運用

1. IT システム間で多くの API が直接連携していたが、今後さらに IT システムが増加すると、API のパフォーマンスやセキュリティ制御において、API それぞれの情報を網羅的に把握することが困難になり、適切なコントロールができなくなる懸念が

あった。そのため、API ゲートウェイの管理ツールを導入し、API 連携は IT システム間で直接連携せずに API ゲートウェイを経由する方式に変更することとした。その結果、API ゲートウェイの管理ツールを利用することで API のパフォーマンスやセキュリティを一元管理することが可能となり、全ての API を適切にコントロールできるようになった。

- ① IT システム全体の処理能力や可用性を高めるために、API 設計時に、API を利用する IT システムの可用性に応じて、データをやり取りする送信側と受信側の IT システムの状況・要件を考慮してパフォーマンス定義を厳密に行った。また、API 運用時は、API のパフォーマンスが想定以上に必要になる状況に備え、業務別に API ゲートウェイのサーバを分けた。さらに、想定パフォーマンスに近づくことで、スケールアップあるいはスケールアウトするよう自動化した。その結果、API の処理能力による問題は発生せず安定稼働している。

また、技術的な取り組みだけでなく、以下の取り組みを併せて行った。

- ② 社内外のメンバーが開発する API が、セキュリティや開発要件を満たしているかを統制するため、チェックリストを用意した。チェック項目全てに合格してはじめて API アクセスキーを発行するといったワークフローとしての意味も持たせて運用した。そのような取り組みの結果、セキュリティや開発要件に問題を発生させず、安定した IT システム運用を実現することができている。
- ③ 外部サービス提供 API を利用する際、外部サービス提供 API が利用不可となった場合を想定し、自社システムの可用性に応じた SLA (Service-Level Agreement) を締結した。幸い、外部サービス API が利用不可になることは、調査時点では発生していない。
- ④ データの整合性を確保するため、一つのトランザクションは、一つの API で処理するように設計した。その結果、トランザクション処理におけるデータ不整合の問題は発生していない。

②品質向上

1. API ゲートウェイを構築し、API 管理ツールのコネクタ機能(データベースなどへの接続に関するコードの複雑さを軽減する仕組み)を活用することで、IT システム間連携を構築する際、接続仕様が統一されたため、安定した品質を保つことができた。また、接続仕様を都度検討する必要がなくなったため、開発スピード向上にもつながった。
2. API ゲートウェイを経由しない API のバージョンアップを検討する際は、バージョンアップ対象の API に接続している IT システムの調査にかかっている。そのため、基本的には、API のバージョンを上げずに対応している。API のバージョンを上げる対応が必要になる場合は、影響する IT システムとタイミングを合わせて、バージョンアップを行っている。このように、API ゲートウェイを経由しない

API のバージョンアップ作業は煩雑になるため、なるべく API ゲートウェイを経由するよう取り組んでいる。そうすることによって、API 管理ツール上で普段から API に接続している IT システムを管理できるため、API のバージョンアップに伴い影響を受ける IT システムの調査時間を短縮することに成功した。

③データ活用促進

社内の様々なビジネス部門が自ら最適なデータ分析を可能とするため、部門ごとにデータ活用分析基盤を構築した。データ活用分析基盤上に、社内外のデータを整理して保管するために API を活用している。データ活用分析基盤に、API を介して必要なデータを整理したことで、各ビジネス部門でのデータ活用が進んでいる。

(3)小売業 C 社

<役割> API 管理者、API 開発者

<概略>

API 中間層を土台にして、IT システム運用の異常事態にも対応できる備えを準備し、新しい技術や設計思想によって、ビジネス変更のスピードや顧客の期待に対応した。

<背景・課題・ニーズ>

時代と共に顧客の期待が常に変化し、求めるスピードが速くなっていく中、小売業として顧客やビジネスの変化やスピードに、柔軟かつ素早く対応することが必要になった。そのため、現行システム(B)と競争領域の独自アプリケーション(A)の顧客向けシステムの間 API 中間層を構築して、競争領域の独自アプリケーション(A)の顧客向けサービス機能の新規開発や継続的な改善による IT システム変更、サービスを安定的に提供するための非機能要件担保を吸収する形にした。この API 中間層を中心に、API の活用が増えていく中、急激な API 利用増加など、異常事態にも対応できる安定した IT システム運用が課題になった。また、ビジネスの変化のスピードや顧客の期待に対応するために、API を活用した具体的な取り組みを推進した。

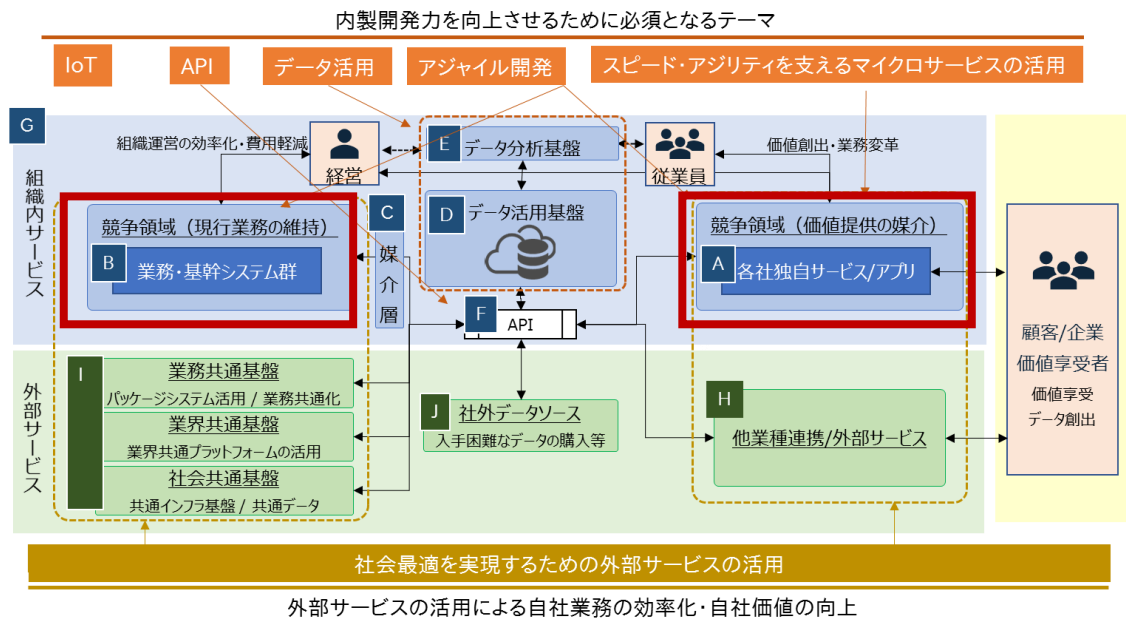


図 4.5.10 本事例の API を活用したスサノオ・フレームワーク領域

<取組みとその効果>

① 異常事態にも対応できる安定した IT システム運用

1. API 中間層に API ゲートウェイを構築して、API を集約し、管理ツールや性能監視ツールを導入して、API の利用状況を可視化できるようにした。可視化によって見えるようになったデータを元に、競争領域の独自アプリケーション(A)の顧客向けシステムで急激な利用増加が発生した場合の、他 IT システムへの影響の波及経路や利用量を予測することが可能になり、事前に流量制御設定を行うことや問題が発生した場合の想定ケース・対応策を準備しておくことで、実際に問題が起こった場合に、適切なアクションを即座に実行することが可能になった。
2. 外部サービスが提供する API も活用しているが、外部サービス提供会社側の問題により、API が利用できなくなるリスクがある。外部サービス提供会社と共に問題を防ぐための対策を行ったが、完全に問題を無くすことはできないため、問題が発生した場合の備えを強化した。性能監視ツールを導入して、API を利用する IT システム全ての担当者が、問題を即座に見発見・共有できるようにした。そうすることで、問題が発生した場合の初動を早めることができ、また、外部サービスを利用する IT システムとその IT システムに関する IT システムの担当者も、問題の発生状況や影響を把握しやすくなり、IT システム間で対応を素早くとることが可能となった。

② ビジネス変化のスピードや顧客の期待に対応するための非同期 API 技術活用やヘッドレスコマース設計

1. 多彩な商品を取り扱っているが、現行システム(B)の基幹システムの機能制限により取扱商品に制限が発生していた。例えば、店頭と EC サイトで共通の在庫として扱っている 1 点ものの商品や、予約販売などの店頭お渡し品など、通常商品とビジネス上の取り扱いが異なる商品である。このような制限なく様々な商品を取り扱えるようにする必要があったが、現行システム(B)の基幹システムを変更するには、多大な時間がかかり、ビジネスが求めるスピードで IT システムを提供することができなかった。そのため、多彩な商品の取り扱いに制限が発生しないよう、商品特性に合わせた IT システムを別途構築した。図 4.5.11 のように、それぞれの商品特性にあわせて別途構築した IT システムが、現行システム(B)の基幹システムのデータベース上で該当する商品特性に合致するデータが変更になった場合、非同期 API を介して、該当するデータの変更のみを検知する仕組みにした。従来は、基幹システムの改善が必要であったが、これが不要になったことで、機能提供スピードの改善とコスト削減が実現できた。

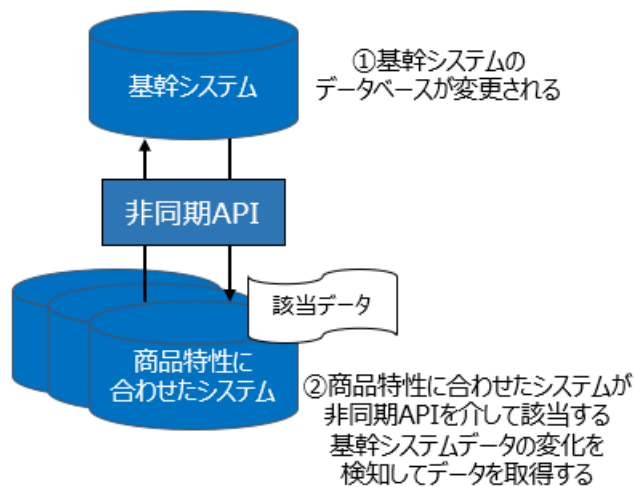


図 4.5.11 商品特性に合わせた IT システムによる非同期 API を活用したデータ取得

この仕組みを活用して顧客の期待に対応した具体的な事例は、店頭と EC サイトで共通の在庫として扱っている 1 点ものの商品である。1 点ものの商品は、店頭で購入することができるが、近くに店舗がない顧客は、そのような商品を購入することができなかった。企業は、EC サイトを運営していたため、全国の顧客に販売することができたが、店頭と EC サイトで在庫を共有した場合、EC サイトでの注文情報を店舗側に伝えるまでに数時間

のタイムラグがあった。そのため、その数時間の間に 1 点ものの EC サイト販売商品が、店頭で売れてしまうリスクがあるため、在庫を共有することが難しかった。そこで、図 4.5.12 の商品特性に合わせた IT システムによる非同期 API を活用したデータ取得の仕組みを活用した。図 4.5.12 のように、顧客が店頭陳列販売商品を EC サイトで注文した場合、注文データが基幹システムに登録される。商品特性に合わせた IT システムが、非同期 API を介して基幹システムに登録された「EC サイトと店頭で在庫を共有する商品」の注文データを取得する。その後、「EC サイトと店頭で在庫を共有する商品」の注文データを元に在庫確保指示データを作成し、店舗側の物流システムへ送信する。これに気付いた店員は、店頭陳列販売商品をすぐに EC サイト販売商品として確保する。このようにして、以前の仕組みであったら顧客が EC サイトで注文してから、店員が在庫確保指示を受け取るまでに数時間のタイムラグが発生することになっていたが、数十分に短縮することが可能になった。基幹システムと物流システムのデータ連携のタイムラグを短縮することで、EC サイトで販売した商品が、店頭で買い物をしている顧客に購入され、商品在庫が無くなってしまいうリスクを低減することができた。顧客にとっては、店舗が近くになくても、1 点ものの商品を EC サイトで購入できるようになり、企業にとっては、店頭陳列商品を EC サイトでも販売することで、EC サイト専用在庫を持つ必要がなくなり在庫回転率を向上させることができた。

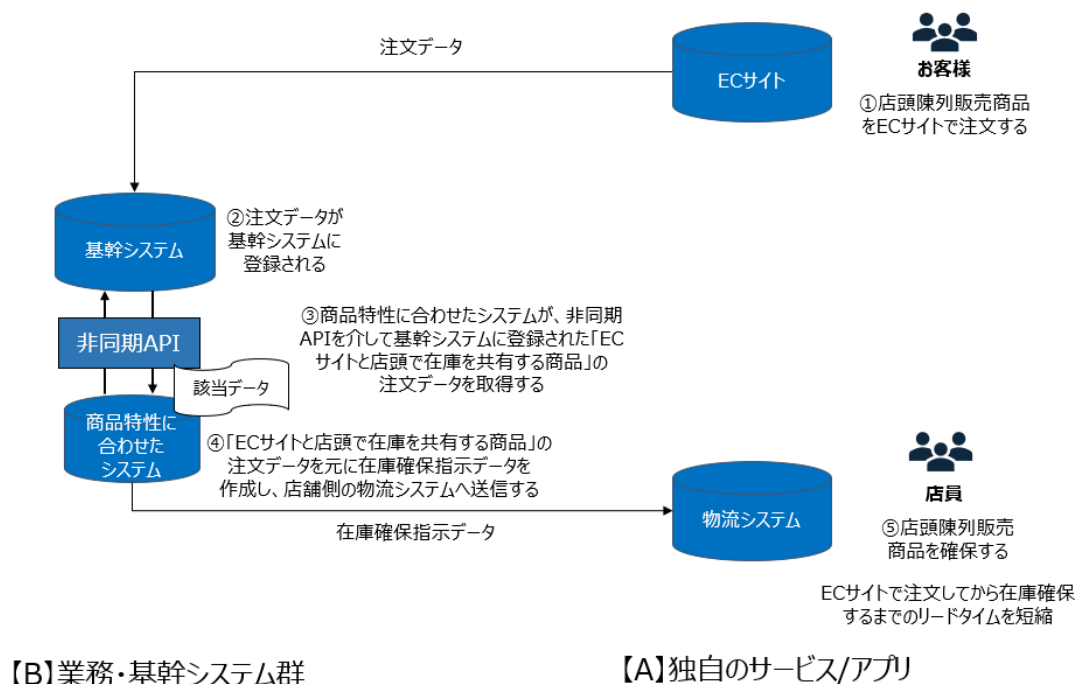


図 4.5.12 顧客が店頭陳列販売商品を EC サイトで注文した際のデータの流れ

- 競争領域の独自アプリケーション(A)の顧客向け IT システムは、顧客や業務の変化に対応していく必要がある。独自の EC サイトを展開しているが、特に変化が激しいため、変化に柔軟に対応するために、商品管理や在庫管理など社内業務を担うバックエ

ンド部分と顧客が直接触れるフロントエンド部分を分離して開発し、バックエンド部分とフロントエンド部分を API 連携する「ヘッドレスコマース」の設計思想でサービスを構築している。その結果、バックエンド部分は改修せずに、フロントエンド部分のみを改修するなど変更箇所を限定させることができた。

(4)製造業 D 社

<役割> API 管理者、API 開発者

<概略>

API ツールの機能を幅広く活用して開発・運用の品質と生産性を大きく向上させ、API を活用してデータと業務プロセスを連携させ業務変革を推進した。

<背景・課題・ニーズ>

社内のデータや業務プロセスを連携させ、社内業務を統合することで、業務の効率や生産性を向上にするための取り組みを推進していた。社内外の様々なデータや業務プロセスを連携させていく上で、API はキーとなる要素の 1 つであると考え、API を活用してきた。基本的に自社 IT 部門で IT システム開発・運用を行っていたが、スサノオ・フレームワークの現行システム(B)、データ活用基盤(D)、データ分析基盤(E)、外部共通基盤(I)、社外データソース(J)の領域で、多くの IT システム間の連携を API で行うようになる中、ビジネス部門から IT システム開発・運用の品質を落とさず生産性を向上させていくことを常に求められた。そのため、API 開発・運用の品質向上と生産性向上を図ることが必要となった。

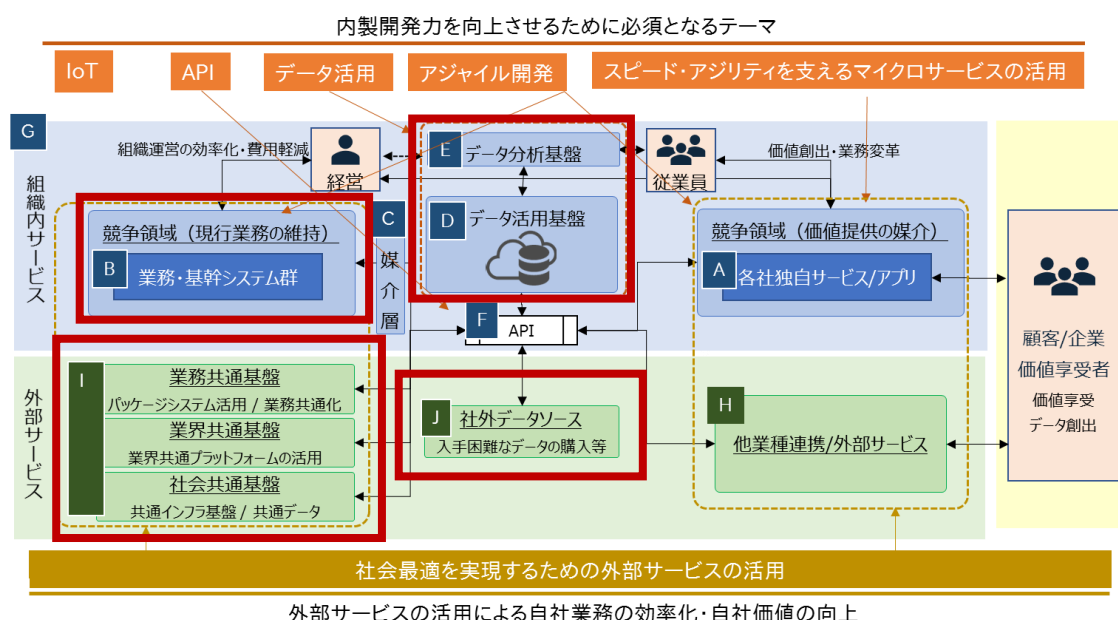


図 4.5.13 本事例の API を活用したスサノオ・フレームワーク領域

<取組みとその効果>

データと業務プロセスを連携させ業務変革を推進するために、IT 部門は、ビジネス部門から常に、品質を保ちつつ開発サイクルを短縮することを求められた。例えば、顧客へ提出する見積書作成業務の品質向上・作成の効率化を行うため、製品仕様の選定ミス防止や適切な価格・納期を顧客に提示することを実現させる必要があった。製品仕様コードを基に社内の IT システムに保存されている製品仕様やカタログプライス等のデータを取得することで製品仕様の正確性を担保し、顧客体験を向上させ受注件数を増加させることをビジネス部門から期待された。そのようなビジネス部門からの業務変革ニーズに短期間で応えるために、IT システム間連携において、API 管理ツールを導入した。API 管理ツールが提供する以下の様々な機能を活用して、品質向上と生産性向上を実現した。

①API 管理ツール活用による開発の品質向上と生産性向上

1. API コネクタ機能（データベースなどへの接続に関するコードの複雑さを軽減する仕組み）を活用して、ツールが提供する標準的な接続方法を何千という接続に適用したことで標準化が進みデータベースなどへの接続作業工数を下げつつ、品質を向上させることができた。
2. API カatalog自動作成機能（API を作成した際に、自動的に API がカatalogとして一覧化され、各 API の情報を調べることができる仕組み）を活用して、API 利用が増える中、API カatalogを通じて API 情報が共有できるようになり、API 利用者が情報を入手・調査するための時間を短縮できるようになった。
3. API 一覧機能（どの API をどの IT システムで利用しているかを表示することができる仕組み）を活用して、API を改修・追加する際、改修・追加する API を利用している IT システムを容易に把握できるようになった。その結果、API の改修・追加を行う際は、影響する IT システムを漏れなく同時に変更できるようになった。
4. API ノーコード開発機能（コードを書かなくても API エンドポイント(URI)などを開発できる仕組み）を活用して、API を開発している。ノーコード開発ができるようになったことで、以前と比較して開発工数を削減することができた。また、IT 部門の新人が、ノーコード開発を行っており、ベテランと同様の品質で開発できている。

②API 管理ツール活用による運用の品質向上と生産性向上

1. API プロダクト機能（個別の API ごとにアクセス権やトラフィック制限をしなくても、同じアクセス権やトラフィック制限を用途に合わせて事前に登録しておき、個別の API は事前に登録されている用途別の設定を利用できる仕組み）を活用して、API トラフィック制御、IP アドレスのホワイトリスト作成、API 認証を行った。機能を活用することで、パフォーマンス制御や現行システム(B)のバックエンドシステムの負荷軽減、セキュリティ管理を一元的に行うことができるようになり、業務の生産性も向上した。

上記のように API 管理ツールを導入し、様々な機能を活用した結果、開発工数は、取組み前と比較して大幅に削減することができた。また、管理ツールを活用して一元管理できるようになったことや、業務を自動化して属人化させないようにしたことで、開発・運用品質を大きく向上させることができた。

4. 6. IoT

4.1.6 でも示しているように、「スサノオ・フレームワーク」と IoT とは、データ活用基盤を接点として関連を持っており、また、DX を実現する上でも、IoT の重要性が増してきている。

本項では、そうした IoT の活用による効果や事例などについて、説明を行う。

4. 6. 1. IoT の概要

IoT (Internet of Things) とは、インターネットなどのネットワークにコンピュータ類のみならずセンサーやカメラ、工作機械や家具などさまざまな「モノ」が接続され、データを収集したり相互に情報をやり取りしたりする概念を指す。1999 年に IoT を提唱した Kevin Ashton によれば、コンピュータが RFID (Radio Frequency Identification) やセンサーを用いて「モノ (Things)」から迅速かつ正確に情報収集を行うことにより、省力化とともに、自らが世界を観察、特定、理解するようになる概念 (狭義の IoT) であった。しかし、現在の IoT は、離れた場所にある多くのデバイスが生成するデータをリアルタイムで取得することが可能となったことから、収集した莫大なデータ (ビッグデータ) を用いて新しい知見を得たり、リアルタイムに機器やシステムを制御したりすることも重要な特長 (広義の IoT) となっている⁵⁸。

なお、利用目的によらず、企業内部の重要なデータや、顧客・利用者のプライバシーに関わるデータを収集する場合には、IoT に求められるセキュリティとプライバシー保護の水準が高まる点に注意が必要である。また、IoT によって、1) 機器等の想定しないうつながりが発生する、2) 適切に管理されていないモノもつながる、3) 身体や財産への危害がつながりにより波及する、4) 問題が発生してもユーザーにはわかりにくい、といったリスクが指摘されており、つながる機器やシステムの安全安心対策の必要性が指摘されている⁵⁹。

4. 6. 2. IoT 活用の背景

ここでは、こうした IoT が活用される背景として、DX との関わりについて、説明する。

⁵⁸ IPA : DX 白書 2021, 2021 年 10 月 11 日. https://www.ipa.go.jp/ikc/publish/dx_hakusho.html

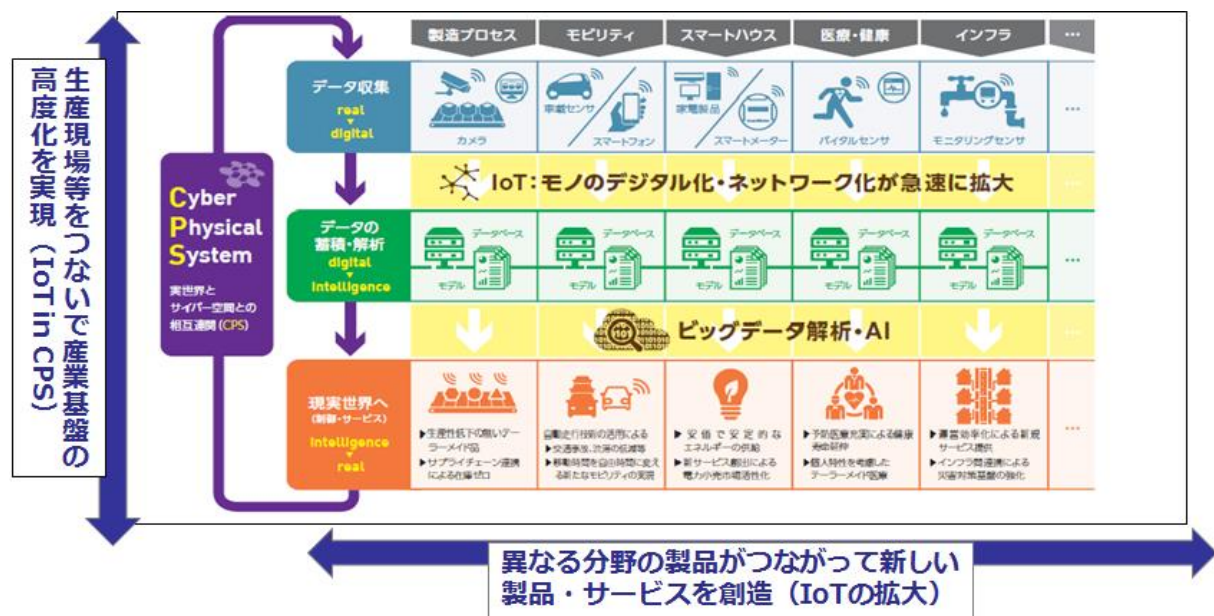
⁵⁹ IPA : つながる世界の開発指針(第 2 版), IoT の安全・安心の確保に向けた仕組みの構築.
https://www.ipa.go.jp/sec/our_activities/iot.html

DXについては各所で様々な定義されているが、経済産業省のDX推進ガイドライン⁶⁰における定義によれば、“D”（データとデジタル技術）を活用して“X”（製品やサービス、ビジネスモデル、及び業務、組織、プロセス、企業文化・風土の変革）を行うことである。この定義での手段である“D”が（広義の）IoTということになる。

IoTを活用したDXの実現に関連する技術として、Cyber Physical System (CPS)を挙げることができる。経済産業省 産業構造審議会は2015年、中間とりまとめ⁶¹において、産業基盤の高度化を図るCPSのイメージを公表している。同図には、各分野におけるCPSが横連携することでビッグデータ解析等により新たな価値を生み出すというイメージが整理されている。すなわち、各分野においてIoTとビッグデータ解析・AIによりCPSを構成するが、この垂直型のCPSがIoT及びサイバー空間によってさらに横連携することにより、そこで新しい価値が生まれるというものである（図4.6.1）。

また、総務省 情報通信審議会も2017年、第三次中間答申の概要⁶²において、IoT総合戦略：基本的枠組みとして、IoTにより現実世界で収集されたデータ（ビッグデータ）がサイバー空間でAIにより分析されて現実世界にフィードバックするデータ主導社会（Data Driven Society）の概念を示している（図4.6.2）。

これらはいずれも、内閣府が提唱する、サイバー空間とフィジカル（現実）空間を高度に融合させたシステムにより、経済発展と社会的課題の解決を両立する、人間中心の社会（Society 5.0⁶³）を目指すものである。Society 5.0では、データの生成・収集・流通・分析・活用を徹底的に図ることによって、製造過程はもとより、あらゆる社会経済活動を再設計し、社会の抱える課題解決を図るとしている。これは、社会の変革（トランスフォーメーション）と言える。



⁶⁰ 経済産業省：デジタルトランスフォーメーションを推進するためのガイドライン（DX推進ガイドライン）を取りまとめました，2018年12月12日。 <https://www.meti.go.jp/press/2018/12/20181212004/20181212004.html>

⁶¹ 経済産業省：「情報経済小委員会 中間とりまとめ報告書」，産業構造審議会 商務流通情報分科会，2015年5月21日。 https://www.meti.go.jp/committee/sankoushin/shojo/johokeizai/report_001.html

⁶² 総務省：「IoT/ビッグデータ時代に向けた新たな情報通信政策の在り方」（平成27年諮問第23号）に関する情報通信審議会からの第三次中間答申，2017年1月27日。 https://www.soumu.go.jp/menu_news/s-news/01tsushin01_02000216.html

⁶³ 内閣府：Society 5.0, https://www8.cao.go.jp/cstp/society5_0/index.html

図 4.6.1 社会全体が CPS により変革されるデータ駆動型社会⁶⁴

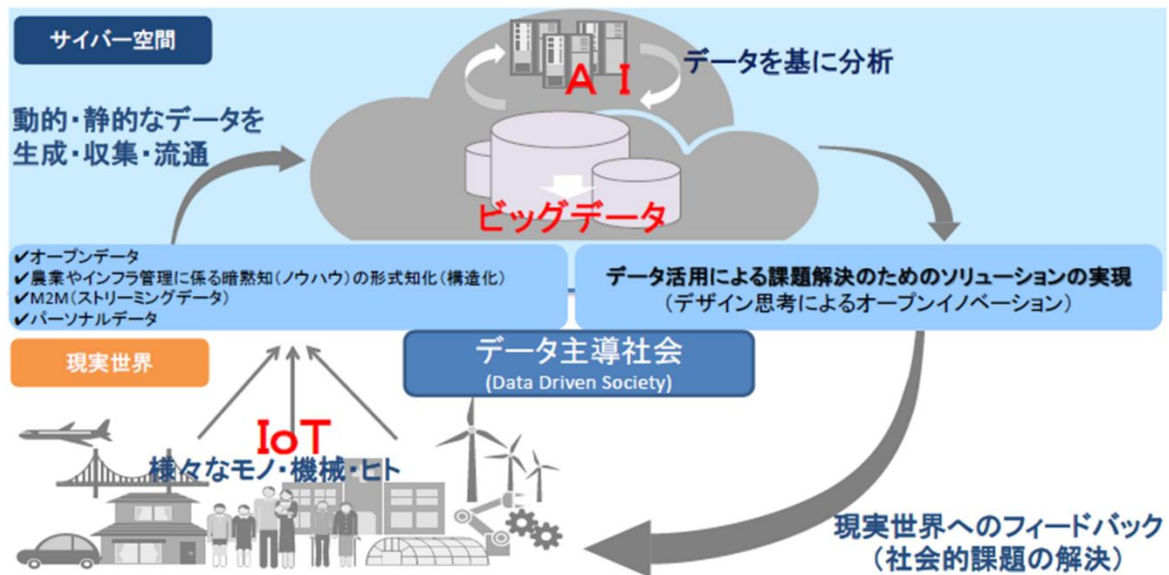


図 4.6.2 ビッグデータの利活用を鍵とするデータ主導社会

4.6.3. IoT の活用により実現できること

DX を実現する上でも欠かせないこうした IoT は、普段の生活の中や日々の業務の中で活用されてきている。

身近な消費者向けの IoT 活用例としては、ネットワークに接続した家電やスマートスピーカー、スマートウォッチなどが挙げられる。スマートフォンやスマートウォッチを用いて、外出先から家電を操作したり、心拍数や運動量を計測してアプリで分析したり、自宅の機器、家族やペットの現在の状態を確認したりといったことは今日では珍しくない。IoT は、離れた場所にある多くのデバイスが生成するデータをリアルタイムで取得することを可能としている。

IoT において収集したデータはクラウドに蓄積されることが多い。しかし、収集したデータの処理をすべてサーバ上（クラウド上）で行うのではなく、データを収集した末端の IoT 機器やその近辺（エッジ）で行うエッジコンピューティングもよく使われる。エッジコンピューティングを用いる主なメリットとしては、通信に伴う遅延を低減してリアルタイム性を改善したり、インターネットやクラウドの障害時でも処理を継続したりできることと、エッジで生データを処理し、安全または必要なデータのみをサーバへ送信することにより、セキュリティを高めたり、通信データ量を軽減して通信コストやストレージのコストを下げたりできることが挙げられる。

IoT はまた、消費者向けサービスだけでなく産業でも活用されている。AI やデジタルツインなど、高度化するデータ分析・シミュレーション技術の基盤となるデータ収集技術として、企業の規模や業界を問わず国内外で導入が進んでいる。製造業における IoT の活用では、工場内の生産設備や製品の状態の収集のように、業務プロセスの中で IoT を用いて生産性向上や業務プロセス

⁶⁴ 前掲「情報経済小委員会 中間とりまとめ報告書」の図をもとに加筆

の最適化を図る場合や、IoT 家電のように製品をインターネットに接続して利用状況を把握し、製品・サービス改善による顧客価値向上のために情報収集を図る場合がある。

生産性向上や業務プロセスの最適化を目的とした活用の場合、生産物や設備の状況、周辺の環境などのデータをセンサーで取得し、通信を介して IoT ゲートウェイやプラットフォームなどを経由したのち、クラウドや中央のサーバでデータの分析・シミュレーションを行うというシステムが一般的である。IoT を用いることにより、企業は大量のデータを取得可能になる。取得した大量のデータに対し、AI などのデータ分析技術を用いて高度な分析・予測・シミュレーションを行い、その結果を生産設備やプロセスなどにフィードバックすることにより、コスト削減・生産性向上に寄与することができる。また、ネットワークに接続された機器のデータなどから得られた知見に基づいて異常発生を予測して機器を停止するなど、離れた場所から機器をリアルタイムに監視・制御・操作することも可能になる。最近では、製造業のみならず農業・医療・建築・流通といった幅広い業界で IoT が活用され、業務プロセスの効率化や製品品質の向上といったビジネス上の価値を産みだしている。

製品・サービス改善による顧客価値向上を目的とした活用の場合には、製品に IoT センサーを搭載し、製品の状態や、顧客による製品の利用状況・利用頻度などのデータを収集する。得られたデータを分析することにより保守サービスの品質向上や各顧客向けにカスタマイズしたサービスの提供が可能となり、顧客への提供価値を向上させることができる。センサーによる自動収集データ以外に、コールセンターに寄せられる顧客からのクレームデータを組み合わせて分析することもある。近年、顧客の行動や従業員の判断など、ヒトに関するデータが収集対象となっていることは重要なトレンドである。新型コロナウイルス感染症の拡大後、都心部の人流データが盛んに報道されるようになったように、個人の端末を経由してデータが収集されている。ウェアラブル端末などから取得した健康状態のデータをもとに個人の生活習慣の改善を勧めるといった医療・ヘルスケア業界の例のように、企業の顧客データなどから得た知見を個人にフィードバックし、助言・注意を行う IoT の使い方も注目されている。

このように IoT は、センサーから得られた情報を可視化するシンプルなものから、予測・最適化といった知的な処理まで行う技術的に高度なものまで、幅広く活用されている⁶⁵。

4.6.4. デジタルツイン

デジタルツインは、IoT と密接な関係にある技術である。デジタルツインとは、物理空間と対になる双子（ツイン）を仮想空間上に構築し、モニタリングやシミュレーションを可能にする仕組みを指す。物理空間と仮想空間、及び両者の情報連携によって構成されるデジタルツインにより、物理世界から得られたデータを用いて仮想空間上にリアルタイムに、あるいはライフサイクルとしてモノやヒトなどが再現され、現状分析や将来予測などが可能となる。

工場生産ラインの変更や都市計画など、実空間で実物・模型を作成してシミュレーションし試行錯誤を重ねるには多大なコストや時間が必要となる場合がある。そこで仮想空間上に作成した

⁶⁵ 前掲 DX 白書 2021

デジタルツインを用いてシミュレーション・分析・最適化を行い、その結果を物理空間へフィードバックすることにより、コストや時間を短縮し、設計や生産などの効率化・改善が可能となる（図 4.6.3）。たとえば、物理空間からのデータを用いて仮想空間に再現された製品に対して、仮想空間でのシミュレーションにより、場合によっては物理世界とのリアルタイムなデータ交換を行いつつ、当該製品の設計・製造や使用に関わるパラメータについて様々な値の中から最適な値を導いた上で、物理空間においてその値を使った実物で最終確認することにより、試作・確認の時間を大幅に短縮することができる。時間とともに変化する対象や環境の状態を仮想空間内のデジタルツインに反映する際には、物理空間からのデータ収集が必要となるため、IoT センサーによるデータ収集が必要不可欠である。

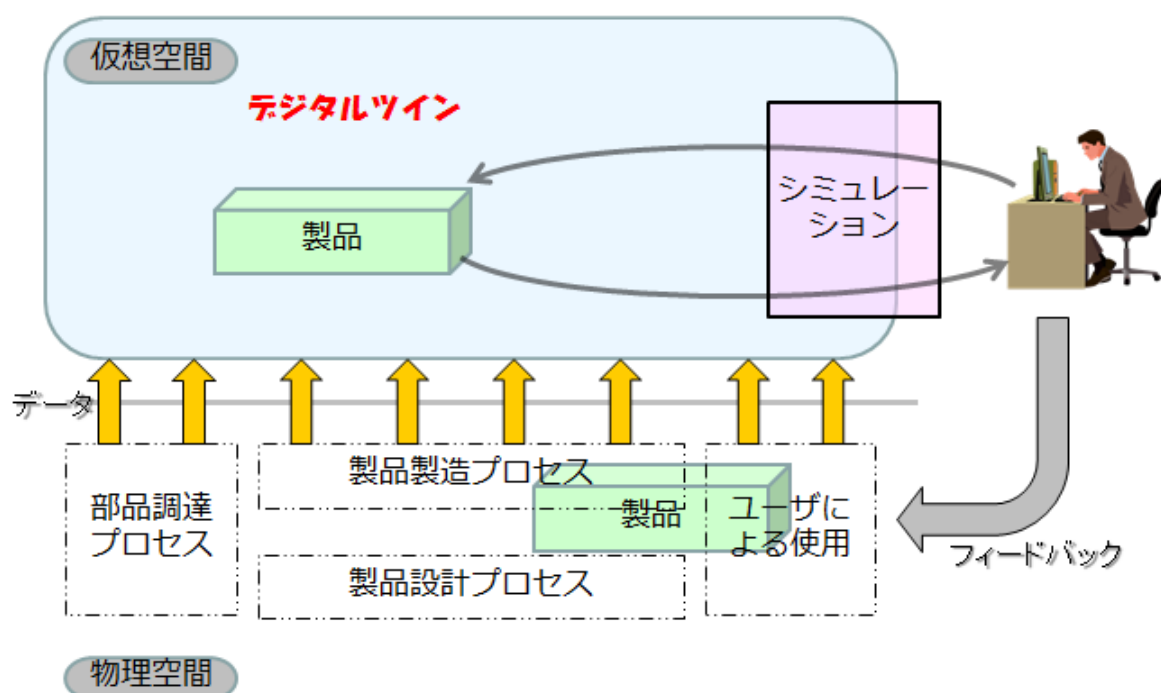


図 4.6.3 デジタルツインの活用例

なお、現在、関心が集まっている技術・サービスとしてメタバース⁶⁶があるが、デジタルツインをもとにメタバース内にその製品・サービスを再現し、参加している人々にそれを評価してもらうという、仮想ユーザー評価というべき応用も構想されている。これは、単なる物理現象の次元を超えた仮想検証を可能にするものとして、デジタルツインのメリットを一層増大することが期待される。

⁶⁶ メタバースとは、仮想 3D 空間に多くのユーザーが参加し「現実」としての営みを行うことを指す。90 年代に SF 小説「スノウ・クラッシュ」に登場した概念で、00 年代には典型的な実装例である「Second Life」がサービスインするなど、けして新しい技術ではないが、VR デバイスなどの技術進化の中で、Facebook 社が今後の技術的発展方向として本腰を入れたことで新たに注目が集まっている。

4.6.5. 製造業での IoT の活用事例

ここでは、中小規模の製造企業が IoT を活用して DX 推進に取り組んだ事例⁶⁷を紹介する。

(1) 計器を IoT 化して市場を開拓（株式会社木幡計器製作所）

1909 年に創業した株式会社木幡計器製作所は、圧力計業界や保全業務の将来に不安を抱いて新規事業の開拓に取り組んでいた中で、2014 年、計器の遠隔監視が可能な「IoT 圧力計」を企業間共同プロジェクトで開発した。これは、従来の機械式圧力計に無線デバイスを搭載し、計測結果をクラウドサーバに送信するものである。なお、この製品はリチウム電池で駆動できるように低消費電力の設計がなされており、電源が供給されない環境でも使用できる。

その後 2017 年には、IPA「先進的 IoT プロジェクト支援事業」の支援を得て、既設アナログ式計器を IoT 化した「後付け IoT センサー・無線通信ユニット」を開発した。工場の製造現場や大規模商業ビルの機械室などの孤立した環境の設備には、数多くのアナログ式計器が既に取り付けられているが、この開発により既存の計器を新たな IoT 計器に取り換えることなく、各機器の稼働状態を遠隔から監視できるようになった。

同社は、2019 年から、ガスボンベディーラーとの協業により、医療機関向けに「IoT 圧力計」を利用した医用酸素ガスの残量監視システム「メディカルガスモニター®」を導入し始めた。

(2) 情報の可視化とデジタルデータの蓄積（株式会社 IBUKI）

1933 年に木型の製造販売企業として創業した株式会社 IBUKI は、その後、射出成形用金型の設計・製造を行っていた 2014 年、製造業コンサルティング業を行う株式会社 O2 の傘下に入った。以降、同社の支援の下で、工場全体の就業管理や営業管理などの IT 系のデータと工場のマシンの OT 系のデータなどの蓄積データを用いて、データの管理を一括して行うエンジニアリングサービスを展開していた。

そうする中で、本業の金型製造において、金型内へセンサを内蔵し、センサー値をデータ化することにより金型の中の状態を可視化する「IoT 金型」を開発した。内蔵するのは、射出された樹脂の熱変化を監視するための温度センサー、金型内部の樹脂圧力を監視するための圧力センサー、射出時に発生する金型の型開き挙動を監視するための変位センサーである。

「IoT 金型」により、射出成形の量産中における金型の不具合の可能性を事前に察知できたり、樹脂の変化が値によって表されメンテナンスの必要性を予測できたりする。また、圧力や温度といった数値的な比較も可能となり、その差異を読み解くことにより、より実成形に近い状況の予測ができ、成形不良の未然対策実施や量産立上げリードタイム短縮へとつながる。

さらに同社は、デジタルデータ蓄積・管理と「IoT 金型」とを結び付け、射出成形における熟達者の判断の AI 化も試行している。これは、「IoT 金型」の成型で内部の状況をデータと

⁶⁷ IPA：中小規模製造業の製造分野におけるデジタルトランスフォーメーション(DX)のための事例調査報告書，2020 年 7 月 20 日， <https://www.ipa.go.jp/ikc/reports/mfg-dx.html#section8>

して把握できるようになったために可能となったことであり、従来、勘に頼る部分があった最終調整の修正回数の削減を実現する。

(3) 自社の強みを徹底的に追求しサービスビジネスを構築（碌々産業株式会社）

1903年に機械工具類の輸入販売業として創業し、後に汎用のマシニングセンタを製造するメーカとなった碌々産業株式会社は、1996年には同社の強みを生かした事業への転換が必要と判断し、ミクロン以下の加工が可能な「微細加工機」を他社に先駆けて開発した。その後、2010年には、オペレータが経験と勘に頼らなくても微細変位の補正を行えるように、複数のセンサーにより機械の挙動を見える化する M-Kit を開発した。さらに 2018 年には、ユーザーの支援を新たなビジネスモデルと捉え、ユーザーと一体となって微細加工機を遠隔監視し、予防保全、早期トラブル解決を実現する「AI Machine Dr.」を開発した。

微細加工の精度には、周囲の温度や湿度の変化が影響を与える。しかし、どの程度の温度や湿度の変化が製品のパフォーマンスにどの程度の影響を与えるかの見極めは専門性が高く、ユーザーが理解することはほとんど不可能である。同社の機械では、あらゆる部位にセンサーが装着され、温度や電力、流量、圧力、負荷、時間、指令等の 36 項目のデータを最小 10 ミリ秒の間隔で取得・可視化し、製品に付属するメモリに記録するとともに専用のクラウドにアップロードする。それを同社の専門技術者が遠隔で確認し、動作が不良となった場合の原因究明や製品の使い方の指導などを行っている。

「AI Machine Dr.」により、ユーザーの情報に基づいた保守・点検に加え、使い方の指導といったコンサルティングサービスも提供できるようになった。また、ユーザーによる同社製品の使用方法や使用環境をもとに、より使いやすい製品の設計・開発にいかすことが可能となった。

(4) 中小金型メーカの IoT 活用とつながる工場（株式会社ウチダ製作所）

1980年にプレス加工メーカとして創業した株式会社ウチダ製作所は、大手自動車メーカ向けプレス加工部品の製造販売が順調に伸びる中で、繁忙期には人材不足により社内で金型を全て作り切れないため、金型製造を外部委託せざるを得ない状況であった。

このような課題に対応するため、地域の金型メーカと連携して企業連合をつくり、大手・中堅金型メーカの市場であった高難易度プレス金型の製作事業に乗り出している。同社を中心とする企業連合は、IoT や AI などのデジタル技術を活用して「つながる工場」を実現しているため、地場の金型メーカだけでなく、地理的に離れた金型メーカからも提携協力を取り付けている。

企業連合が提供する「金型共同受注サービス」では、参加する金型メーカの製造設備に IoT デバイス取り付け、設備の稼働状況をクラウド上で把握することにより、各金型メーカの仕事を予測し、金型ユーザーからの注文を受けたときに、設備能力と仕事量に応じて最適な受注先金型メーカを選択できる。また、参加する金型メーカは、保有する設備の稼働状況を提供することにより、受注の機会が増加する。参加する金型メーカが増えて、つながる工場全体が仮想的に保有する設備の台数や種類が増えれば増えるほど、空き設備を獲得するチャンスが増えるため、金型ユーザーにとっては、ワンストップで多種の金型を注文できるばか

りか、短納期化も期待できる。

4.7. 社会最適を実現するための外部サービスの活用

本項では、3.1 項で述べた**社会最適**を実現するために、外部システム（外部システム（H）、外部共通基盤（I）、外部データソース（J））を活用する際の考え方を示す。

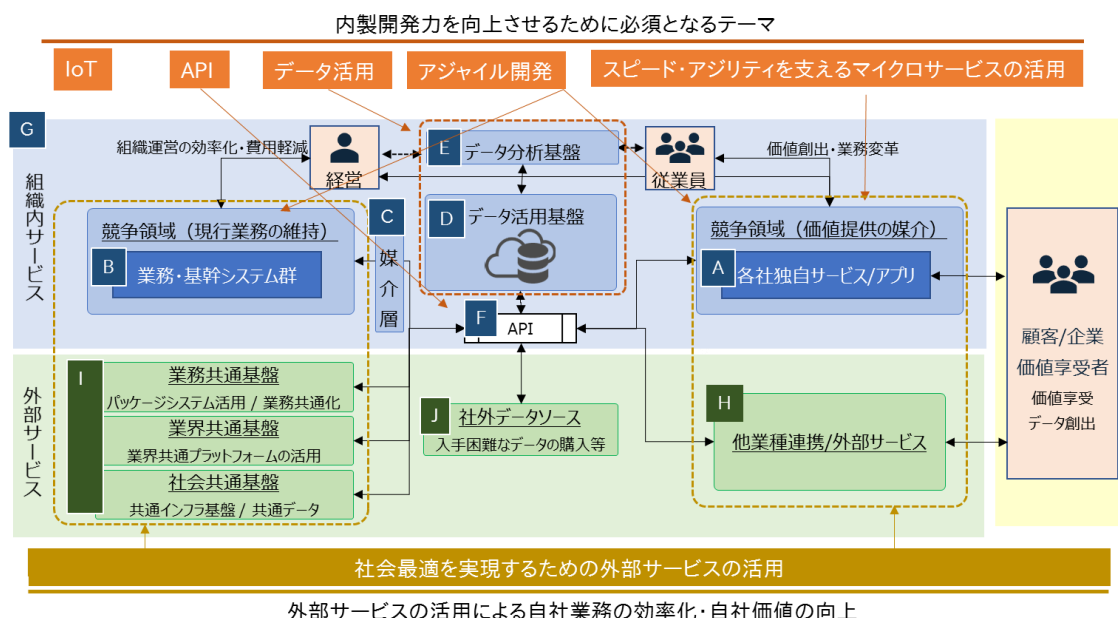


図 4.7.1 あるべき IT システムを実現する技術要素群「スサノオ・フレームワーク」（再掲）

4.7.1. 社会最適とは

社会最適に関しては 3.1 項でも解説しているが、ここで、改めて本書における社会最適という言葉の定義を示す。社会最適とは、「**競争領域、非競争領域を明確化し、非競争領域においてすでに社会に存在するリソース（外部サービス）を活用してビジネスを支えるシステムを構成し、そこで生まれる割り勘効果によって、自社の IT 投資額、開発・保守体制、リスク対策費用・人員を最適化することで、リソースを競争領域へ投下してビジネスを強化すること**」と定義する。つまり、社会全体を最適化するというような壮大な話ではなく、自社のデジタル投資や人材を効率的に活用するために、外部社会に存在するサービスをうまく取り込んだアーキテクチャを構築することを意味する。

4.7.2. 割り勘効果

社会最適の重要な概念である割り勘効果について説明する。システムの開発・運用・保守には機能開発、テストなどの直接的な開発作業に加えて、基盤のバージョンアップ、制度改正、セキュリティ、災害などへの対応など様々な作業が発生する。自社開発の場合、これらを行うための

体制やコストは、それを利用する利用者である企業が負担することになる。同様にシステムに関するリスクも一企業が負うことになる。一方で、ソフトウェアには、複製するコストが極端に小さいという特性がある。この特性を利用して、同じソフトウェアを複数の企業が共通で使ったり、汎用的な外部サービスを利用したりすることで、個々の企業が負担すべき開発コスト、人員、リスク対策コストが低減される。これを割り勘効果と呼ぶ（図 4.7.2）。商用サービスの場合は、必ずしも開発コストを利用者で割ったものが価格になるわけではないが、競合するサービスベンダがいる場合には市場原理として価格が適正化されることが期待されるので、自社開発よりは割安な価格で同等な機能が利用できる。

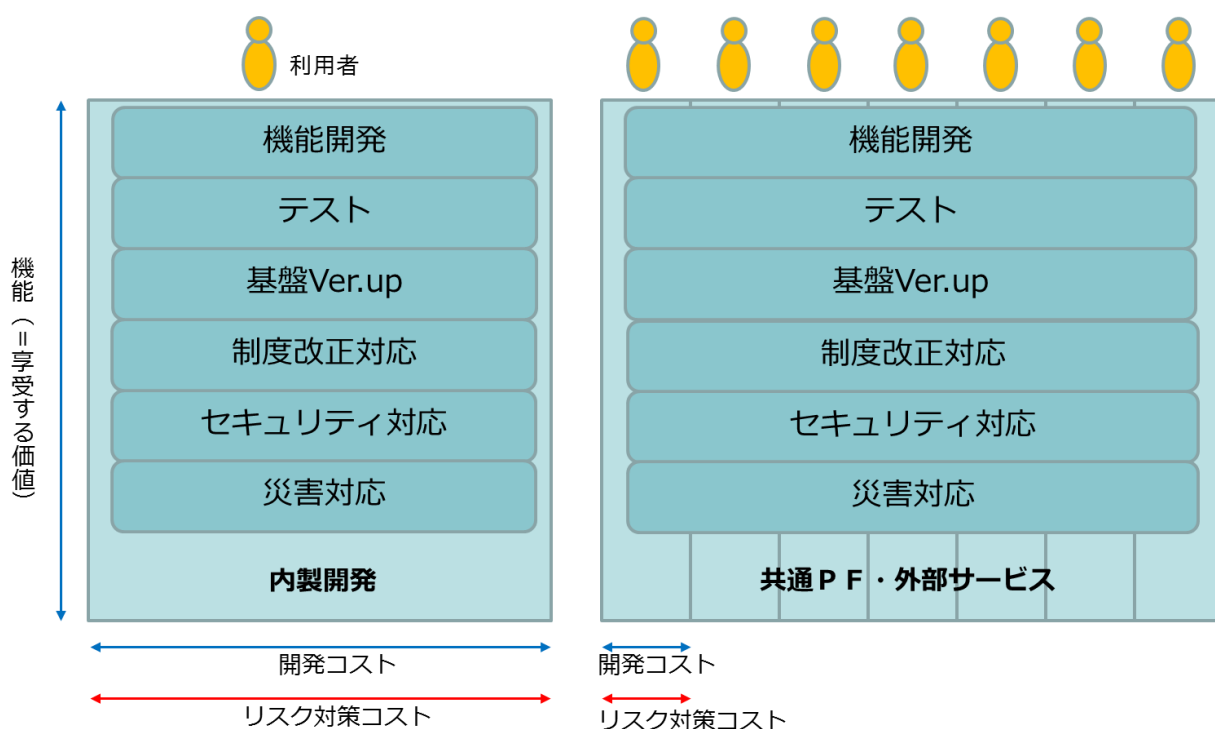


図 4.7.2 割り勘効果のイメージ

4.7.3. 割り勘効果が期待できる外部サービスの種類と特徴

前項で説明した、割り勘効果を享受できる主な外部サービスには以下のものがある。ここでは外部サービスを、SaaS などの実装を隠ぺいして機能を提供するものから、パッケージやコンポーネントなどの実装モジュールを提供するものを含む広い概念で扱う。

(I-1) 業務共通基盤（パッケージソフト、SaaS）

特定のベンダーやサービスプロバイダによって提供される多くの企業で共通している業務をサポートする汎用的なソフトウェアやサービス。一般にオーダーメイドで開発する際に比べて安いコストで導入できる。開発や保守がベンダーやプロバイダによって行われ、ユーザーはその機能を利用するための対価を支払い、ベンダーやプロバイダーが

利益をえる。

(I-2) 業界共通基盤・社会共通基盤（共通プラットフォーム）

特定の業種・業界で共通している業務や、特定地域で共通している業務をそのコミュニティ内で共同開発・運用することで割り勘効果を得る。開発や保守がユーザーのコミュニティやコミュニティの判断で委託された業者によって行われる。コミュニティは原則として利益を得ない。

(H-1) 他業種連携サービス

企業間や企業と個人を連携するためのサービス。自分たちではリーチできないようなユーザーや取引先とこのサービスを活用することでつながることができ、バリューチェーンの範囲を広げることができる。

(H-2) 外部サービス（技術コンポーネント）

データ分析や機械学習など高度な技術を実装したコンポーネントを提供するサービス。利用者はこのコンポーネントを自分たちのアプリケーションに組み入れることで、やりたいことを実現する。自前では開発できないような技術や試行錯誤が必要な機能を他社での試行錯誤の実績を利用することで安価に活用できる。

(J) データソース

大規模なデータそのものや、その分析・学習結果を提供するサービス。一企業だけでは入手困難なデータの収集結果や、多くの教師データによって学習されたモデルを割り勘効果で活用することができる。

これらの外部サービスを有効に活用することによって、社会最適を実現することができる。

4.7.4. 外部サービス利用のメリット・デメリット

前項までで、社会最適を実現するために、割り勘効果を期待できる外部サービスについて解説したが、これらの外部サービスを活用する際のメリットとデメリットについて改めてこの項で解説する。

【メリット】

- ・ 割り勘効果によって、開発・保守のコスト・リスクが抑えられる
- ・ 開発・保守の人員を張り付ける必要がない
- ・ 法律や制度改正に個別に対応する必要がない
- ・ 基盤のバージョンアップ等の対応を個別に行う必要がない
- ・ 既存のサービスを利用する場合はすぐに利用を開始できる

【デメリット】

- ・独自の業務に対応した細かいカスタマイズが難しい
- ・契約の仕方によっては自社のノウハウが外部に流出する
- ・自社独自の業務の変更に迅速に対応することが難しい
- ・ブラックボックス化により、障害やセキュリティリスクの対応が長期化する可能性がある
- ・サービスプロバイダの都合でサービス料金の値上げやサービス終了される可能性がある
- ・既存業務や既存サービスとのすり合わせが必要

このような、メリット・デメリットを考慮して、組織内で利用するそれぞれのシステムを内製するか、外部サービスを利用するかを検討して、組織のシステムアーキテクチャを検討していく必要がある。次項にて、システムが担う業務の特性による、外部サービスの利用の考え方を示す。

4.7.5. 競争領域、非競争領域と外部サービス

前項で述べたメリット・デメリットを考慮した、外部サービスの利用方法を考えるにあたって重要な概念に2つの競争/非競争性がある。組織における業務の単位で2つの競争性を見ることで、システムとしてどのような形で外部システムを活用すべきかが見えてくる。

【事業的競争性】他社と同様な業務を行っていて、その事業そのものや扱っている情報が競争における差別化要因になっているかどうか

【技術的競争性】デジタル技術やソフトウェア技術として他社に比べて特徴的な技術を使っているかどうか（特許化されたコア技術を保持している。他社に比べて性能や精度が高い技術を保持しているなど）

この2つの競争性を縦横にマトリックスにしたものを図 4.7.3 に示す。このマトリックスの領域ごとに、外部システムの活用方法を考えていく。ここでは、いわゆる IT ベンダーやサービスプロバイダではなく、彼らからユーザー企業と呼ばれる IT やデジタル技術を本業としない組織を想定して議論を進める。ベンダーやサービスプロバイダとの関係は 4.7.6 節にて述べる。

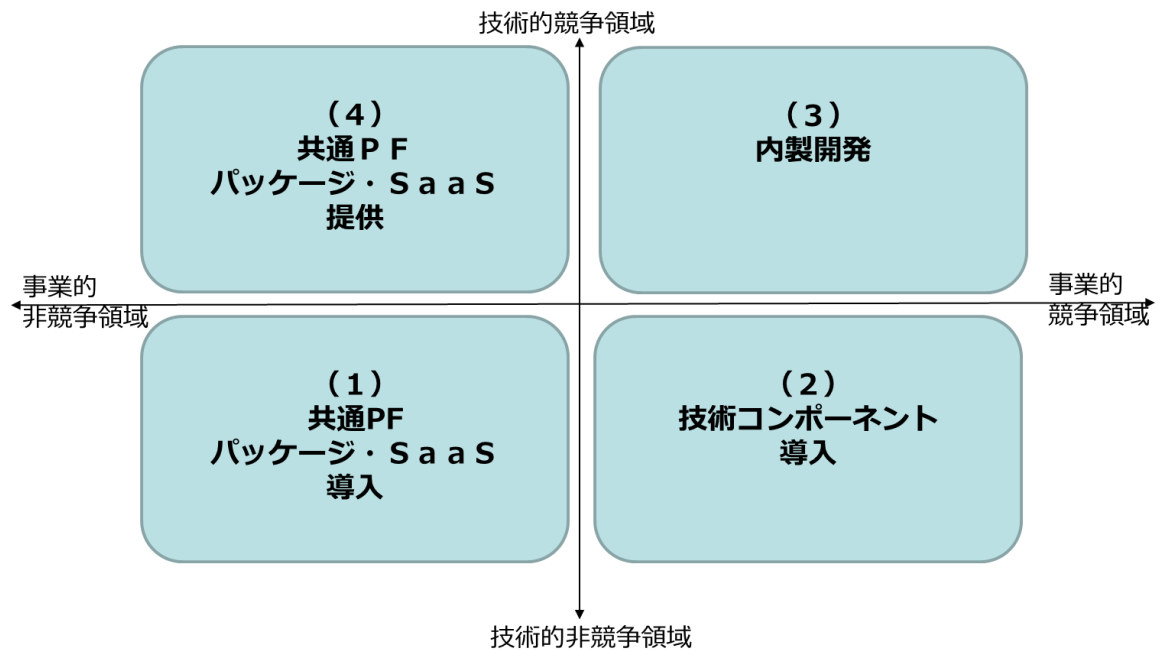


図 4.7.3 2つの競争性

(1) 事業的非競争領域×技術的非競争領域

事業的にも、技術的にも競争要素がないこの領域（図 4.7.3 左下）の業務を支えるシステムは、独自で開発するコストや負うべきリスク対策コストを割り勘効果によって軽減することによって、競争すべき領域に投資やリソースを充てることができる。業務単位でシステム化を実現するパッケージソフトウェアや SaaS を活用するか、業界内での共通化を目指す共通プラットフォームとそのプラットフォーム上で共通化されたアプリケーションを活用することが望ましい。ここで注意しておく点として、この領域の業務を一括して一つのサービスプロバイダに丸投げしてしまうと、業務がブラックボックス化されてしまう可能性があることが挙げられる。そのような状態になってしまうと、より効率的に業務をサポートでき、より安価なサービスが登場した際にそれを選択することが困難になってしまうといういわゆるベンダーロックの状態になることがある。そのため、それぞれの業務間のインタフェースを自社でコントロールできることが望ましい。

(2) 事業的競争領域×技術的非競争領域

事業的には競争性があるが、技術的な競争性がないこの領域（図右下）の業務を支えるシステムは、競争性のある業務で扱う情報を生み出したり活用したりする部分と、その情報をデジタル技術で処理をする部分に切り分ける。そして、前者を社内で内製（図右上の領域）し、後者をデジタル技術のエンジン（技術コンポーネント）として外部のものを導入する。これにより、競争力の源泉となる情報を社内に囲いながら、先端のデジタル技術の恩恵を受けることが可能になる。

(3) 事業的競争領域×技術的競争領域

事業的にも、技術的にも競争性のあるこの領域（図右上）の業務は、情報を外部に出すべきで

はないし、これを実現する技術をもつパートナーとなりうる他社は存在しない。したがって、これを支えるシステムは、内製開発を行い社内の投資とリソースを集中させることで、組織の競争力の更なる強化につなげることができる。ここの内製化対象は、アプリケーションだけでなく組織内で共通利用するフレームワークも含まれる。

(4) 事業的非競争領域×技術的競争領域

事業的な競争性はないが、技術的な競争性のあるこの領域（図 4.7.3 左上）の業務は、他社にも同様な業務が存在しているが、それを効率的に行う特徴的な独自技術を持っていることになる。この領域の業務は内製開発を行い、技術的な優位性を活かして商品の価格低減などの組織全体の事業的な競争性につなげることでもある。さらに、業務の非競争性を逆手にとって、独自技術をパッケージや SaaS、共通プラットフォームの形に汎用的な実装を行うことで、自らの組織が外部サービスのプロバイダーとなることもできる。その場合には「社外の『業務』を支援するサービスを提供する」という新たな（3）の領域の業務が生まれたと考えることもできる。

4.7.6. デジタル産業の企業4 類型との関係

経産省が公開している DX レポート 2.1 によるとデジタル産業の企業は4つの類型に分類できるとされている（図 4.7.4）。

デジタル産業の企業4 類型

経産省：DXレポート2.1より

- デジタル産業を構成する企業は、その特色を踏まえて4つに類型化できる。

デジタル産業の企業類型

① 企業の変革を共に 推進するパートナー	・ 新たなビジネス・モデルを顧客とともに形成 ・ DXの実践により得られた企業変革に必要な知見や技術の共有 ・ レガシー刷新を含めたDXに向けた変革の支援
② DXに必要な技術を 提供するパートナー	・ トップノッチ技術者（最先端のIT 技術など、特定ドメインに深い経験・ノウハウ・技術を有する）の供給 ・ デジタルの方向性、DXの専門家として、技術や外部リソースの組合せの提案
③ 共通プラットフォームの 提供主体	・ 中小企業を含めた業界ごとの協調領域を担う共通プラットフォームのサービス化 ・ 高度なIT 技術（システムの構築技術・構築プロセス）や人材を核にしたサービス化・エコシステム形成
④ 新ビジネス・サービスの 提供主体	・ IT の強みを核としつつ、新ビジネス・サービスの提供を通して社会への新たな価値提供を行う主体

図 4.7.4 デジタル産業の企業4 類型

ここまで述べてきた2つの競争性と外部サービスの活用の考え方にこの類型との関係を図

4.7.5 に示す。

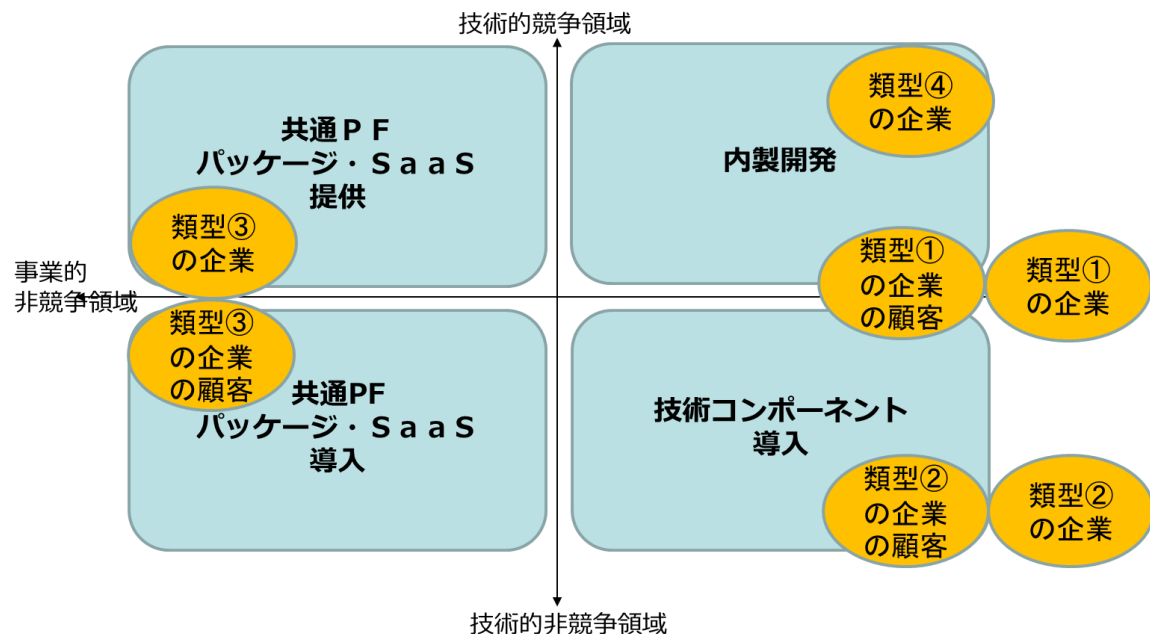


図 4.7.5 2つの競争領域とデジタル産業の企業4 類型の関係

【類型①：企業の変革を共に推進するパートナー】

類型①に属するデジタル産業企業は、パートナーとなる顧客の DX 実現を並走する形で、顧客が求めている変革のゴールを目指す。そのために、顧客が内製開発する部分と外部のサービスを採用する部分の決定や、業務や提供価値の変革をサポートする。そして、内製部分の開発支援も行う。このため、類型①の企業の顧客は図で示した領域に属している。当然のことながら、パートナーとなりうるためには、技術的競争力を持っている必要があるため類型①の企業は図の上の領域に位置づいているべきだが、図の見やすさのために枠外に配置した。

【類型②：DXに必要な技術を提供するパートナー】

類型②に属するデジタル産業企業は、顧客に対して技術コンポーネントを提供し、技術コンポーネントを組み込んだ顧客の DX 実現のサポートを行う。このため、類型②の企業の顧客は図で示した領域に属している。類型①と同様に、技術的競争力を持っている必要があるため類型②の企業は図の上の領域に位置づいているべきだが、図の見やすさのために枠外に配置した。

【類型③：共通プラットフォームの提供主体】

類型③に属するデジタル産業企業は、顧客やパートナーに対して、プラットフォームや業務ソフトウェアパッケージ、SaaS の形で外部サービスを提供する。

【類型④：新ビジネス・サービスの提供主体】

類型④に属するデジタル産業企業は、自らの事業で DX を実現するためにデジタル技術を活用する。技術的競争性のない部分に関しては、技術コンポーネントして外部サービスを採用し、もしくはパッケージや SaaS、共通プラットフォームを利用することで、技術的競争性のある部分の内製や、事業的競争力の源泉となる部分にリソースを集中させて、さらなる競争力をつけていく。

このように、自らの組織の DX を進めて行くためには、該当する業務の競争力によって、組むべき適切なパートナーを選定することが重要である。その前提として、自組織がどのような業務を行っているかを把握して、競争性を判断する範囲を明確にしておく必要があるが、これがない組織も存在するのが現状である。

4.7.7. 競争性の変化

前項までで、2 種類の競争性と外部システムの関係について述べてきたが、ある事業、ある技術が競争領域にあるかどうかは固定的なものではなく時系列で変化する（図 4.7.6）。

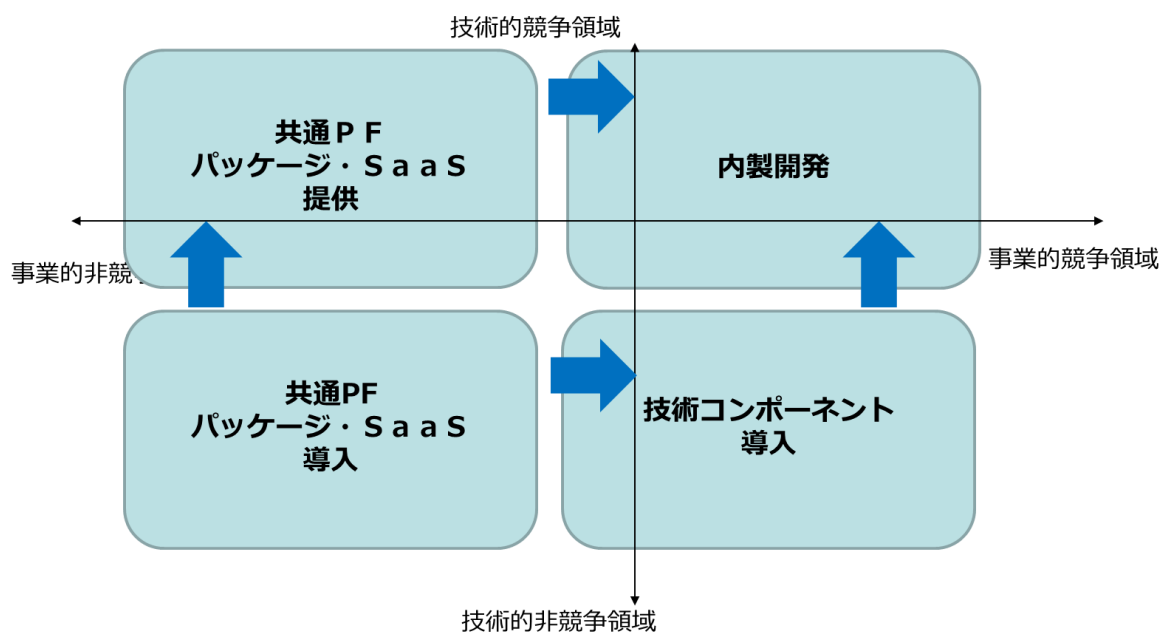


図 4.7.6 競争領域の変化

技術の進歩によって、以前技術的競争性を持っていた技術が、他社でも実装できる技術になる変化（軸が右に移動する。図 4.7.6 右向き矢印）。

業界での合意や法制度、規制などによって業務の標準化がなされた結果、業務の事業的競争性が失われるという変化（軸が上に移動する。図 4.7.6 上向き矢印）

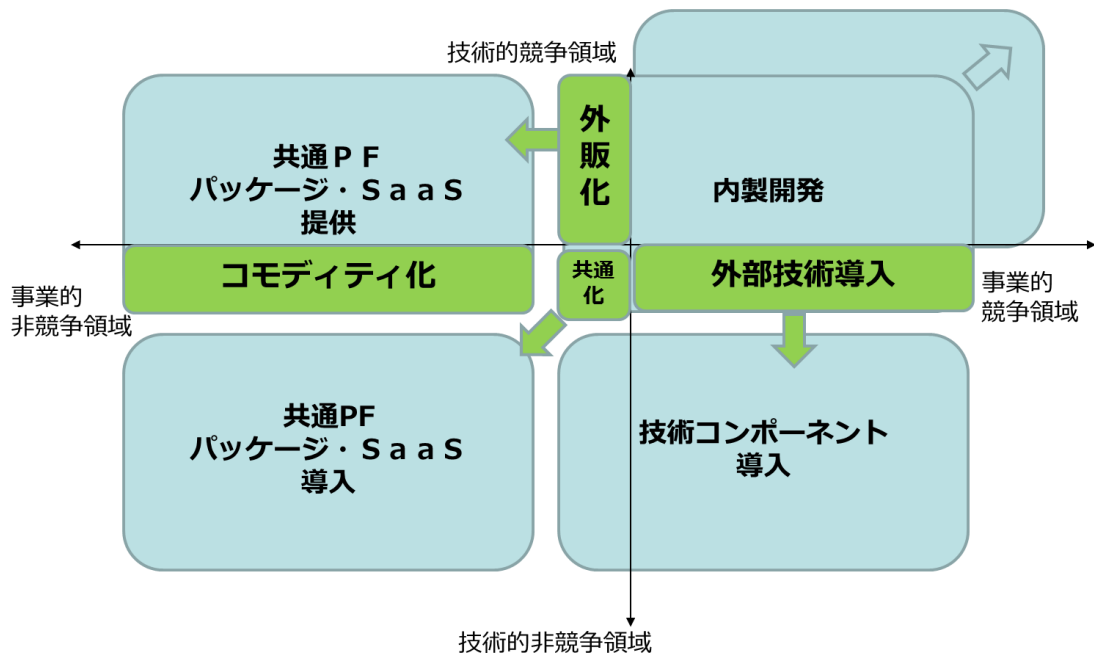


図 4.7.7 競争性の変化への対応

その変化に応じて、外部サービスの活用方法も変化していくべきであり、その様子を図 4.7.7 に示す。

【共通化領域】

事業的にも、技術的にも競争性があるとして内製開発を行っていた業務で、事業的・技術的な状況変化によって、どちらも競争性がなくなった領域。世の中にこの業務を支える効果的なサービスが低価格で提供されているので、業務の標準化を行いそのようなサービスへの移行を行うことで、低コスト化/低リスク化できる可能性があり、リソースを競争性のある領域に集中させることができる。

【外部技術導入領域】

事業的にも、技術的にも競争性があるとして内製開発を行っていた業務で、世の中の技術の進歩により技術的な競争性を失ってしまった領域。内製開発で自社開発している技術よりも高度な技術が外部より提供されているので、そのような技術を技術コンポーネントとして導入することで、事業をより効率的により競争力のあるものにすることができる可能性がある。

【外販化領域】

事業的にも、技術的にも競争性があるとして内製開発を行っていた業務で、世の中の進展により競争性の高かった業務を行う他社が登場してきている領域。技術的な競争性は引き続き保持しているので、それを活用して、低コスト化などで差別化し続けることは可能である。しかし、現時点で競合に見える他社に対して、技術的な競争力のある技術を実装したプラットフォームとそのプラットフォーム上で共通化されたアプリケーションを提供することで、プラットフォーム

として事業のステージを変革させられる可能性がある。

【コモディティ化領域】

事業的な競争性がなく、技術的な競争力があるということで、他社に対してプラットフォームを提供していた業務で、より強い技術を持つ競合他社による類似サービスが登場することにより、技術的競争性を失ってしまった領域。しだいに、技術ではなく価格で競争するという、いわゆる「コモディティ化」が進む。そのため、高いシェアをもっているなど非技術的な競争優位性を持っているかなどを勘案して事業継続性の判断をする必要がある。プラットフォームを外販しておらず、内製開発のシステムで自社の業務を行っている場合には、共通化領域と同様に、業務標準化および外部サービス化を行うことにより、リソースをより競争性につながる領域にシフトできる可能性がある。

ここまで述べてきたように、業務やサービスの置かれている状況は時系列で刻々と変化していく。今までの説明は「内製開発」「導入」など開発時の表現を用いてきた。しかし、ここまで述べてきた考え方は、開発時だけでなく、サービス運用時にも定期的に検討を行って、状況変化に応じたリソースの再配分を行うべきである。

また、ここで示したことは、競争、非競争の領域が連続的に変化する局面において有効な考え方である。いわゆるディスラプションと呼ばれるような、今までの業務の考え方が一変するような変革が起こった場合には、各業務の競争性を一から見直さないといけないこともありうる。そのような事態に対処するためには、外部サービスを利用する領域に関しても、自社でその業務ロジックと業務間インタフェースを把握しておく必要がある。

4.7.8. 外部サービス活用方法のまとめ

今まで述べたことから、外部サービスを活用する方針をまとめる。

- ・ ソフトウェアの複製コストが低いという特性を生かした割り勘効果を享受せよ
- ・ 割り勘効果を実現する外部サービスの種類とその特性を理解せよ
- ・ 自社の業務ごとに、事業の競争性、技術の競争性の有無を判断し、それによって、外部サービスの活用方針を決定せよ
- ・ 競争性は時系列で変化するので、定期的に再評価し、その結果によって方針を見直せ

4.8. 現行システムからあるべき姿への移行

本項では、あるべき IT システムの中の**現行システム(B)**を対象に、移行の考え方、段階的/一括移行を行うことによる効果、考慮点などについて、説明を行う。

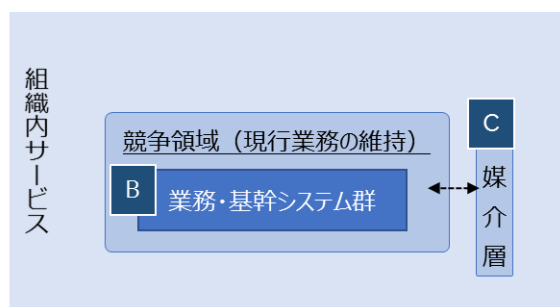


図 4.8.1 あるべき IT システムにおける**現行システム(B)**

4.8.1. 移行の考え方

現行システムに関して、これがいわゆるレガシーシステムとなっており、軽微な修正にも大きな体力が必要となるなど、課題を抱えている例は、少なからず存在すると想定される。なお、本書では、以下の問題を含んでいる現行システムをレガシーシステムとして定義している。

- ① 技術革新がとりこまれず、著しく老朽化している
- ② ブラックボックス化している
- ③ 肥大化・複雑化している（いわゆるスパゲッティ化が起こっている）

この 3 つの問題は相互に関連のある問題である。「①技術面で著しく老朽化している」とは、古い技術要素で構成されているシステムであるため、担当者が退職してしまうとそれに代わるエンジニアが市場にいないという人材の問題や、システムが古くなればなるほど、ハード・ソフトのあらゆる面での保守コスト、そして IT システムの改修が高コスト化する問題を引き起こす。人材不足によって、現行システムの仕様を詳細に把握している人材が組織内にいなくなってしまう、IT システムが「②ブラックボックス化している」状態に陥る。もはや自力での IT システムの改修や、ちょっとした運用改善をすることも難しく、現行維持が繰り返される。また、サブシステム同士が密結合になっているモノリシックアーキテクチャとなっている IT システムを保守・運用し、上記の問題点を抱えながら長年現行維持を継続してシステムを使用し続けると、いわゆるスパゲッティ化を起し、IT システムが「③肥大化・複雑化している」状態になる。改修の際に仕様復元のための現状分析や影響範囲の調査に時間がかかり、さらにコストが必要となる問題を引き起こす。

そうした課題への対応として、まずは、前述した『DX レポート』でも提起されているように、

現状の IT システムを分析し、機能毎に以下の 4 つの観点で評価することが、出発点となる。

- (a)：頻繁に変更が発生する機能は**再構築**
- (b)：変更されたり、新たに必要な機能は**適宜追加**
- (c)：肥大化したシステムの中に**不要な機能があれば廃棄**
- (d)：あまり更新が発生しない機能は**塩漬け**

頻繁に変更が発生する機能(a)については、変更に対して迅速・柔軟な対応を可能とするため、アプリケーションの構造を見直し、新たな IT システムへ再構築を行う対象と考えられる。

また、新たに必要となる機能(b)は、保守性を考慮し、再構築した IT システムへ、適宜追加する対象と考えられる。

評価した結果、不要となる機能(c)があれば、該当の機能に関連する機能やインタフェースへの影響を精査した上で、廃棄する対象と考えられる。

そして、変更の頻度が低い機能(d)に関しては、現状のまま使用を続ける対象と考えられる。

したがって、(c)、(d)を許容しながら、頻繁に変更が発生する機能(a)に対して、再構築するにあたって、現行システムを第 3 章で示した要件を満たす形(=IT システムのあるべき姿)へ移行することが、対応策の 1 つとなるだろう。

その際の移行の考え方としては、大きく 2 つに分けられると想定する。

一つは、現行システムを一度に、IT システムのあるべき姿へ移行する方法であるが、これについては、『DX 実践手引書 IT システム構築編 レガシーシステム刷新ハンドブック』に、現行システム全体の把握の方法等の詳細を記載しているため、そちらを参照されたい。

もう一つは、現行システムから、段階的に IT システムのあるべき姿へ移行する方法であり、本項ではこれを解説する。

4. 8. 2. 段階的移行の方法

現行システムをあるべき姿へ移行するにあたって、移行に要する期間や体制、ビジネス上の方針などから、段階的な実施を方針とした際に、使われる移行手法の 1 つとして、「ストラングラーフィグパターン」⁶⁸がある。

「ストラングラーフィグパターン」では、現行システムから、一部の機能を切り出して、作り変えるということを繰り返し行って、段階的に移行を進めていく。その過程では、現行システムと移行された IT システムが併存する形となる。現行システムを利用する側は、該当の機能が移行された後は、移行後の IT システムへ接続が必要となる。そのため、「ストラングラーファサー

⁶⁸ Martin Fowler, StranglerFigApplication, martinFowler.com,
<https://martinfowler.com/bliki/StranglerFigApplication.html>

ド」⁶⁹と呼ばれる中間層(媒介層)を置いて、現行システムと移行された IT システムに対する接続を「ストラングラーファサード」により振り分けるという対応がとられる。

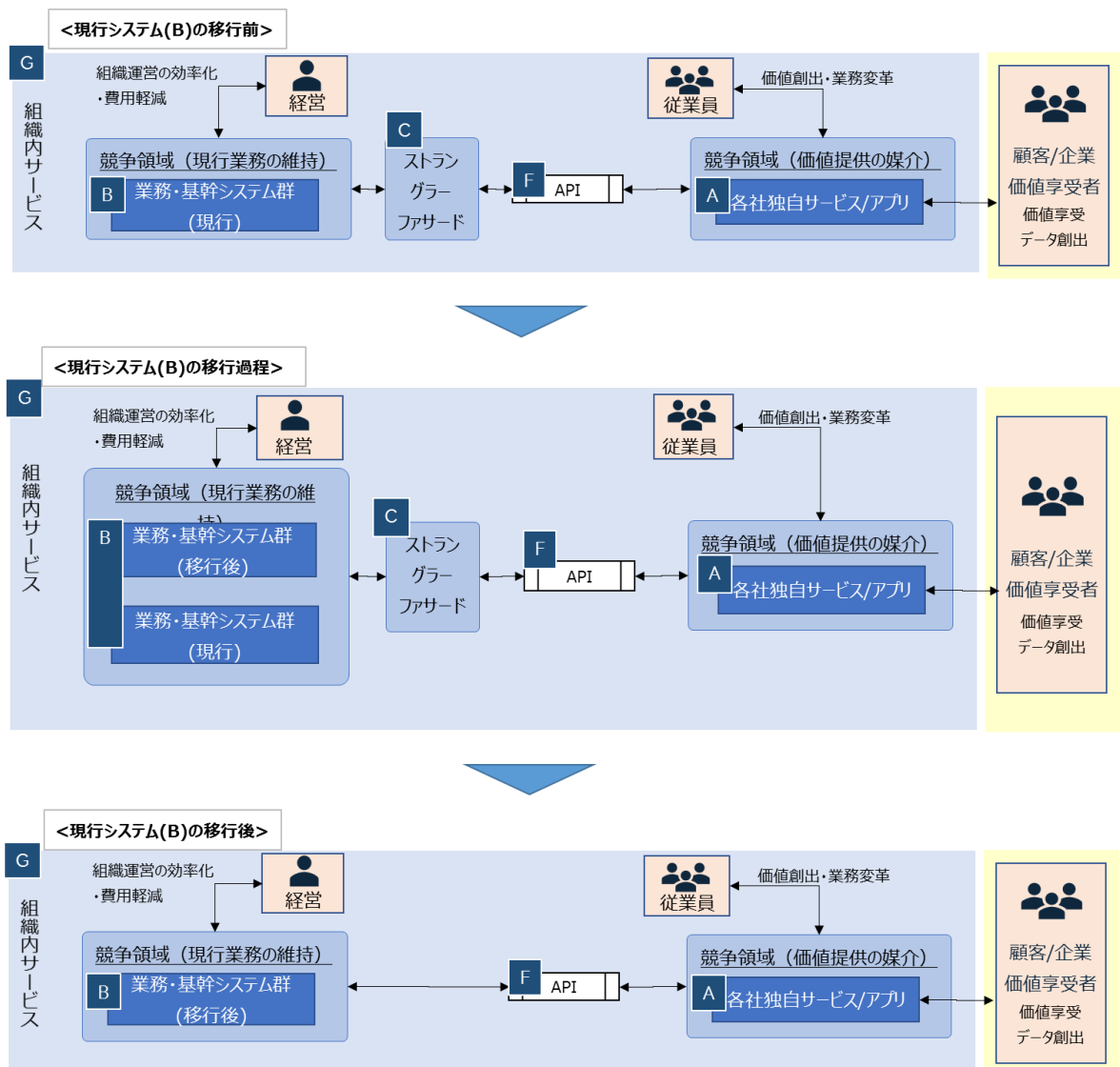


図 4.8.2 現行システム (B) の移行前と移行過程、移行後

4.8.3. 段階的移行により実現できる内容

ストラングラーファサード(C)を活用して、段階的に**現行システム(B)**を移行することで、実現できる内容について、説明する。

①現行システムに対する使用性の確保

⁶⁹ Md Kamaruzzaman, Microservice Architecture and its 10 Most Important Design Patterns, <https://towardsdatascience.com/microservice-architecture-and-its-10-most-important-design-patterns-824952d7fa41>

現行システム(B)は、一部の機能から段階的にあるべき姿へ移行された IT システムへ切り替えが行われていくが、それらの利用者は、ストラングラーファサード(C)を経由し、アクセスを行うことで、現行システム(B)が移行されたか否かを意識せずに、現行システム(B)の利用を継続することができる。アクセスする対象の機能が移行されていれば、ストラングラーファサード(C)によって、移行先の IT システムへ接続が振り分けられ、移行前であれば、従来の現行システムへ振り分けられるということが行われる。

現行システム(B)に対する使用性を損なうことなく、IT システムの移行を進めることが可能となる。

②他の IT システムとの疎結合化

現行システムは、他の IT システムとも連携しているが、そうした連携においても、ストラングラーファサード(C)が、間に入ることで他の IT システムも移行による影響を受けない状態となる。それにより、他の IT システムも、現行システムの移行へ配慮することなく、自身の機能の改修等を進めていくことができる。

4. 8. 4. 段階的移行に関する考慮点

現在の多くの IT システムにおいては、API 技術の活用やプログラムの一部拡張、機能の呼び出しやデータの受け渡しのための外部プログラム実装などにより、機能の一部利用・データの受け渡しなど、ストラングラーファサード手法により可能となってきた。ただ、段階的移行を行うにあたって、現行システムの構造・実装の分析は必要であるし、その結果ストラングラーファサードによる段階的な移行が適さない場合もある。そして、段階的移行の適用が可能と判断される場合も、以下のような点について考慮が必要となる。

<移行の単位>

現行システム側に対する影響を、極力発生させない単位で機能を移管することが、基本的な考え方となる。そのため、例えば、移行対象に関連する機能が複数の現行システムにまたがるようなケースがあれば、関連する機能の単位での移行を考慮する必要がある。

現行システムを単位として、関連する機能の一部だけを移行すると、現行システム内での連携の変更が必要となり、現行システムに対する改修が発生することとなる。

現行システムの機能の関連性を正しく把握した上で、移行機能の単位を確定していくことが求められる。

<並行稼働の実施>

移行実施後、移行前の現行システムの機能についても、稼働を継続させ、移行後の IT システムと並行で稼働させることも、検討すべき事項となる。

並行稼働により、必要なポイントを決めて、移行前/移行後の IT システムの処理結果の突き合わせを実施し、移行後の IT システムの処理内容の検証を行うこともできる。

また、一部の IT システムのみ移行後の現行システムへ接続し、他の IT システムは移行前の現行システムに接続するという状態で、移行した機能の検証を行うことも可能となる。それにより、一定の検証を実施した上で、他の IT システムも移行後の現行システムとの接続へ切り替えるという段階を踏んだ、移行を行うという方法の選択も可能となる。

4. 8. 5. 段階的移行の事例

ストラングラーファサードを活用し、現行システムの段階的な刷新を行っている事例を紹介する。

(1)小売業 A 社

＜背景と段階的移行の概要＞

顧客向けにどのようなサービスを提供できるかが起点としてあり、そのために IT システムの基盤づくりから着手することとなり、以下の 2 つの基盤の整備を行った。

1 つは、「DevOps 基盤」として、開発チームが自律的にインフラ構築や運用監視を出来る環境の整備を行った。そのため、「DevOps 基盤」では基盤に関するガイドラインがあらかじめ整備され、運用に関しても自動化を基本として、構築された。

もう 1 つは、「DevOps 基盤」を土台として整備した上で、顧客向けの新たなサービスを提供する「DX サービス」と現行システムをつなぐ間の基盤として、「ビジネスプラットフォーム」という環境を構築した。「ビジネスプラットフォーム」は、「DX サービス」が現行システムと連携する上で、一元的なアクセスを提供するための基盤として、整備された。



図 4.8.3 A 社の IT システムの関連図

(A 社より提供の資料を元に作成)

現行システムの移行としては、上記の「ビジネスプラットフォーム」をストラングラーファサードとして活用し、段階的に進めている。

現行システムを移行する先の基盤としては、「ビジネスプラットフォーム」を対象としており、「ビジネスプラットフォーム」が、一般的なストラングラーファサードの役割と現行システムの移行先という役割を兼ね備えた形となっている。こうした形態を取っているの

は、「DX サービス」からの現行システムへのアクセスを、より容易にすることを目的としている。それにより、例えば「ビジネスプラットフォーム」上に移行した機能から、「DX サービス」に必要とされる形式で、現行システムのデータを集約して連携するといったことが可能となった。

<得られた効果>

①「DX サービス」のリリースの早期化

「DX サービス」は、基本的に現行システムとの連携が必要となるが、新たな「DX サービス」を構築する際に、「DX サービス」が必要とする形でデータを提供できるように、連携対象となる現行システムの機能の移行を行っている。そのため、新規サービスの提供をより早期に開始するためには、それに合わせた移行の対応を行うことが求められる。

段階的移行により、必要となるタイミングで、必要な機能を対象として、移行を柔軟に実施することができているため、「DX サービス」を短い期間でリリースすることに、寄与することができている。

②段階的移行による開発体力の平準化

現行システムの中の必要な機能から順次、移行を行っているため、移行にかかる開発を一定の体制の中で行うことができている。また、移行先の「ビジネスプラットフォーム」は、「DevOps 基盤」上に構築されているので、移行の開発自体の生産性も大きく向上させることができている。

DX 実践手引書 IT システム構築編

【著作・制作】 独立行政法人情報処理推進機構（IPA）社会基盤センター

【編集責任】 DX推進部 境 真良

【執筆】 DX推進部 三部 良太、田中 秀人、林 航平、鎌田 高輝、
河野 太基、中川 貴之、西本 靖、佐々木 伸一、
佐藤 弥生
産業プラットフォーム部 山下 博之
セキュリティセンター セキュリティ対策推進部
小川 隆一

【専門委員】 名古屋国際工科専門職大学 工科学部 情報工学科 学科長 山本 修一郎
SCSK 株式会社 半田 芳樹

【協力】（企業名 50 音順）

経済産業省 ソフトウェア・情報サービス戦略室

早稲田大学 理工学術院 基幹理工学部 教授 鷺崎 弘宜

独立行政法人情報処理推進機構 羽生田 栄一

独立行政法人情報処理推進機構 アジャイルWG

AGS システムアドバイザー株式会社

島田 寿

株式会社エヌ・ティ・ティ・データ

坂田 祐司、大矢 隆弘

TIS 株式会社

熊谷 宏樹、葛谷 憲彦

株式会社電算システム

八鈕 洋一郎、小西 広朗

齋藤 幸久

東京システムハウス株式会社

比毛 寛之、上野 俊作

株式会社東芝

金子 博

日鉄ソリューションズ株式会社

日本電気株式会社

岸本 一実

株式会社日立製作所

小川 秀人

株式会社日立コンサルティング

佐藤 仁奈

富士通株式会社

御魚谷 かおる、大島 喜芳

橋田 拓志、吉川 明男

三菱電機インフォメーションシステムズ株式会社

小林 敦

Mulesoft, a Salesforce Company

中村 洋介

LINE 株式会社

中村 秀樹

株式会社ワークスアプリケーションズ・システムズ 平原 秀明

(敬称略)

【調査協力】(50 音順)

株式会社エヌ・ティ・ティ・データ経営研究所、ガートナージャパン株式会社、
株式会社日立コンサルティング

【改版履歴】

令和 3 年 11 月 16 日 暫定第 1 版 発行

プラットフォーム変革手引書から、名称変更しました。

令和 4 年 1 月 17 日 暫定第 1.1 版 発行

1.3 を 1.3 と 1.4 に再構成しました。3.1 に小見出しを追加しました。3.2 の用語の統一を行いました。3.2.1 の技術的背景の記載に補足を行いました。

令和 4 年 4 月 4 日 暫定第 2.0 版 発行

2 章を追加しました。(それに伴い、これまでの 2 章⇒3 章、3 章⇒4 章へ変更しました。)1.5,4.1.4～4.1.6,4.4～4.6 を追加しました。

4.2.3,4.3.3 のセキュリティに関する箇所について、加筆を行いました。

令和 4 年 10 月 26 日 完成第 1.0 版 発行

2.4,4.2,4.5.6,4.5.7 を追加しました。(それに伴い、これまでの 4.2 以降を 4.3～に変更しました。)

4.1.4 について、一部加筆を行いました。

【監修所】 独立行政法人情報処理推進機構 (IPA)

【発行所】 独立行政法人情報処理推進機構 (IPA)

社会基盤センター DX 推進部

〒113-6591 東京都文京区本駒込二丁目 28 番 8 号 文京グリーンコートセンターオフィス

URL : <https://www.ipa.go.jp/>

謝辞

本手引書作成にあたり、以下の一覧に挙げた企業をはじめ、多くの企業に先進的な取り組み事例の提供をいただき、ご協力をいただきましたことを、心より感謝申し上げます。

(50 音順)

SRE ホールディングス株式会社
株式会社 EBILAB
株式会社今野製作所
株式会社サンコー技研
株式会社資生堂
住友商事株式会社
住友ファーマ株式会社
SOMPO ホールディングス株式会社
チームシロッコ合同会社
東京海上日動火災保険株式会社
株式会社トライグループ
富士通株式会社
株式会社ベネッセホールディングス
株式会社星野リゾート
三井住友海上火災保険株式会社
株式会社三越伊勢丹ホールディングス
三菱ケミカルグループ株式会社
三菱マテリアル株式会社
Mulesoft, a Salesforce Company
横河電機株式会社
株式会社横浜銀行