

第4節 R&D 部門における効果的なデータ共有・利活用手法と運用体制

(株) キャトルアイ・サイエンス 上島 豊

はじめに

現在の R&D 領域では、データ分析や共有は、極めて属人的な扱いである。客観的なデータ生成、分析が要求される理学、工学領域で、この属人性は大きな問題を孕んでいる。研究というものは創造的な活動であり、個人の才能、発想に起因する「なぜ、そう考えたか？」の部分に属人性が必要なことは当然である。しかし、どのようにデータを生成し、どのように分析し、結論を導いたかは、属人的では問題で、客観的かつトレーサブルであるべきである。実際、センサーや計算機的能力向上により、データの生産性が向上し、扱うべきデータが膨大になり、詳細記録の欠如、偶発的データ取り違い、主観的データ操作が発生する余地が増大し、データ生成、分析プロセスの信頼性が大きく揺らいでいる。

私は弊社を設立する前は研究機関にてコンピュータ、ネットワークの最先端技術を駆使し、自然科学、工学研究を約 10 年間行っていた。その中で R&D データが属人的に処理され、その共有状態がデータの信頼性及び有効活用性を大きく阻害し、共有化及びインフォマティクス分析、AI 化が進まないことを経験した。

本節では、私自身の 10 年の R&D 経験と弊社の 15 年の R&D 支援実績から得た「R&D 部門における効果的なデータ共有・利活用手法と運用体制の作り方」に関して、簡単に解説する。

1. R&D 部門におけるデータ共有・利活用の実情

R&D 部門におけるデータ共有の実情の話の前に、「データ共有」とは、何を意味しているのかを説明したいと思う。「データ共有」とは、読んで字のごとくデータを共有することなのだが、より具体的に言うと、データを生み出した実験及び分析を第三者が再現するために必要な情報を記録し、それを必要な時に迅速かつ確実に参照できる状態に保っておくことを意味する。そういう観点において、ほとんどの R&D 部門におけるデータの共有は「共有」というレベルには達しておらず、単なる蓄積と呼ぶのが相応しいというのが実情である。公的、民間の様々な R&D 部門を見てきた経験をもとに、R&D の現場ではデータがどのように蓄積されているのかを、以下で紹介する。

一人で完結できるような実験や分析では、ほとんどの情報は研究者の頭の中のみであり、注目しているパラメータのみを研究ノートにメモ書きをされているだけのことが多い。実験や分析結果の比較評価がある程度難しい課題に対しては、実験や分析の情報がエクセルに書き写され、比較しやすいように纏められ、個人 PC 内に保存されていることもある。複数の人が関わった実験や分析の場合も上記状況と変わらない部分もあるが、他の人への実験や分析の引き渡し（依頼）に必要な情報のみは、フォーマットが揃えられた用紙もしくはエクセルが準備されていることが多い。また、それら用紙はキングファイルなどにファイリングされたり、エクセルファイルなどのデジタルファイルは共有のファイルサーバなどが準備され、そこに保存するようにされているところもある。これらは、「データを生み出した実験及び分析を第三者が再現するために必要な情報を記録し、それを必要な時に迅速かつ確実に参照できる状態に保っておく」という観点に立つと、「データ共有」ができていないということになる。ただし、研究者本人は何らかの形で頭の中では整理ができており、この状態を「属人的なデータ共有」と呼ぶことにする。

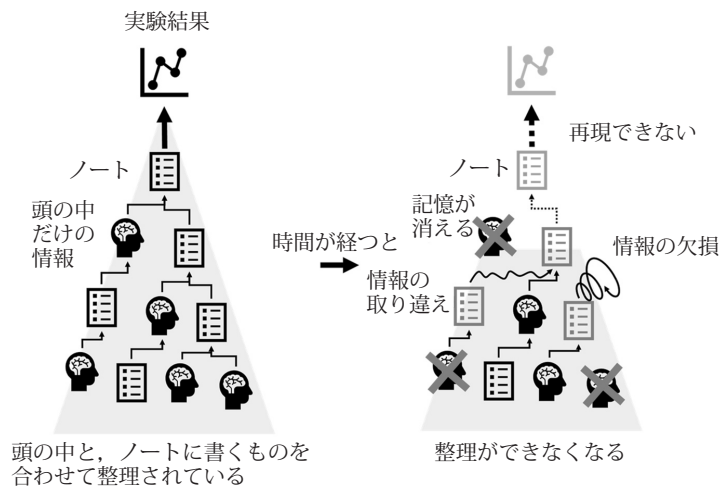


図1 属人的なデータ管理とその特性

データ共有の意識が高い組織では、しっかりしたデータベースシステムを導入し、実験データを蓄積、検索できるようになっているところも稀ではあるが、存在はする。しかしながら、その蓄積されたデータは、まさに蓄積をする以外において利用、活用されていないことが多い。データベースシステムがあり、運用され、データが蓄積されているのに、データの利用、活用が進まないということは、実質的には「データ共有」ができていないということになる。この状態を「形骸化したデータ共有」と呼ぶことにする。

2. 属人的なデータ共有、形骸化したデータ共有状況から生まれる問題点

前項で、説明した通り、ほとんどの R&D 部門では、「データを生み出した実験及び分析を第三者が再現するために必要な情報を記録し、それを必要な時に迅速かつ確実に参照できる状態に保っておく」ことは達成されておらず、「属人的な、もしくは形骸化したデータ共有」状況になっている。本項では、「属人的な、もしくは形骸化したデータ共有」状況がどのような問題を生み出すかについて論じる。

「属人的なデータ共有」状況では、実験及び分析の詳細なことは、実際の実施者しかわからない状況になる。当然のことながら実施者以外の人が実験条件や結果内容を知ろうとした場合、実施者にそれらを聞くしかない。実施者から実験及び分析を第三者が再現するのに十分な情報を提供してもらえれば問題はないのだが、実験データをそういうことができるような状態で蓄積している研究者はほとんどいない。そもそも、どれだけ整理好きの研究者でも、「〇〇の実験・分析情報、データ」が欲しいと言われたところで、実施者として該当するデータを探し出すこと自体、相当困難なことが多いはずである。該当するデータが漏れなく、間違いなく提供されることは、ほぼ不可能と考えてもいいかもしれない。このような状況の中、データの授受を行なうと、間違った情報を基に実験や分析を進めることが発生し、間違った結論が導かれたり、検討を進めたのちに始まりに立ち返って、再実験や再分析を行なわざるを得なくなることもある。

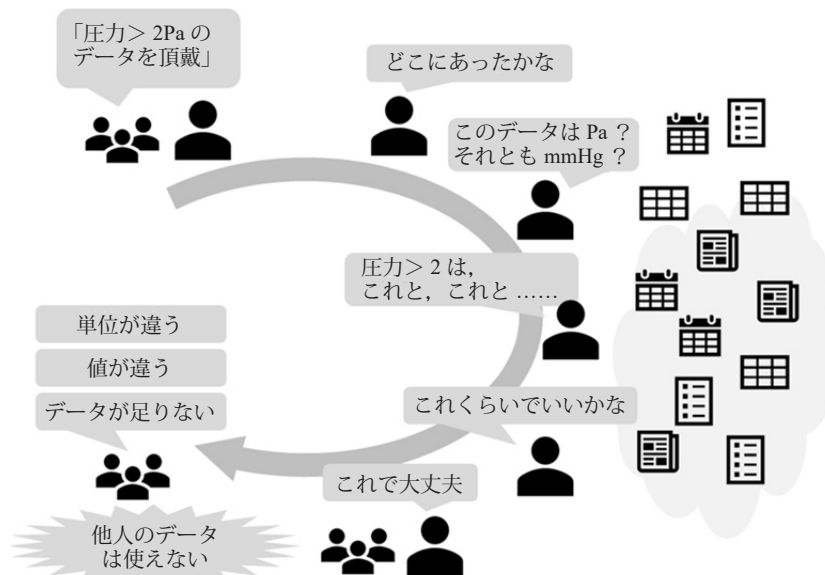


図2 属人的に管理されたデータが引き起こす問題

そういう事態に何度か遭遇すると、研究者同士の信頼関係が崩れたり、他の研究者のデータを参照せず、自分が実施した実験や分析のみしか参照しないような状態が進行していく。このような状態は、研究自体の属人化を加速し、研究の蛸壺化、継承困難化を進めてしまう。

「形骸化したデータ共有」状況は、利用、活用が形骸化するという問題だけでなく、利用、活用されないこと自体も問題である。どんなデータ蓄積システムでも構築自体にコストがかかり、運用、メンテナンスにもコストがかかる。また、そこにデータを登録することも研究者に余分な作業を発生させることになる。それだけコスト、労力をかけても、利用、活用されないのであれば、システム化しない方が良いのではないだろうか？システムが構築され、それが稼働していると、それだけで何かが良くなったと思いがちだが、研究効率という観点ではシステム化以前より悪くなっていることもあるのである。実際に、このような「実験及び分析を第三者が再現するために必要な情報が記録されていない」システムが多いのである。システムが利用、活用されていないので、蓄積されたデータに問題があることなどの発覚が遅れ、大量の不完全なデータ蓄積、つまり、役に立たないデータ蓄積に終わってしまうことが多いのである。したがって、「実験及び分析を第三者が再現するために必要な情報が記録されている」にも関わらず蓄積されたデータが利用、活用されていないのであれば、速やかに利用、活用を促す仕組みづくりを見直す必要がある。

3. 属人的なデータ共有、形骸化したデータ共有状況が生み出される原因

前項では、「属人的な、もしくは形骸化したデータ共有」状況が、どのような問題を生み出すかについて説明した。本項では、そのような問題があるにもかかわらず、なぜ「属人的な、もしくは形骸化したデータ共有」状況が生み出されるのか？、その原因に関して考察する。

「属人的なデータ共有」状況が生み出される直接的原因は、記録が必要な項目及びそのフォーマットが決まっていないからである。R&D部門の業務は、新しい実験及び分析方法は日進月歩で生まれていくため、固定的に項目やフォーマットを決めても、すぐ使い物にならなくなる。したがって、「属人的なデータ共有」状況が生み出される真因は、「R&Dの変化に追従できる項目及びフォーマット」と、その「利用徹底を遂行できる運用体制」がないことということになる。「変化に追従できる項目及びフォーマット」は、ある程度技術でカバーできるが、「利用徹底を遂行できる運用体制」というものは、なかなか難しい問題を孕んでいる。

実験及び分析を行なう場合、実験及び分析の全てのパラメータを変化させることは、まずない。目標とする特性、性能値を達成するためのいくつかの主要因を想定して、ごく限られたパラメータのみを振って、実験及び分析を行

なうはずである。その場合は、その他のパラメータは一定に保つものであり、いちいち記録に残さなくとも頭の中に残っているので、記録を省略することが多いのである。当然、実験及び分析の最中や直後は、記録がなくとも実施者自身であれば実験及び分析の再現は可能である。目の前の実験及び分析の結果に意識を集中しすぎるあまり、その時はその他の一定パラメータは忘れるわけがなく記録するまでもないと感じてしまうので、記録することを省略してしまうのである。つまり、「変化に追従できる項目及びフォーマット」があったとしても、記録する、しないに属人性が入り込んでしまい、「実験及び分析を再現するのに十分な情報」が記録されない状態になってしまうのである。また、そもそも実験及び分析の最中の研究者は、実験及び分析パラメータを忘れるわけがないという意識があり、「実験及び分析を再現できるだけの情報記録」をしていないその状態をそんなに悪い状態ではないと認識をしている研究者が多いことも属人性に拍車をかけている。

そもそも人間、というか生物一般は、論理的に物事を考えるよりも、多数の経験をすることで、パターンマッチング的な認識を形成することが得意なように作られている。人の顔を覚えたり、物の名前を覚えたり、歩行や自転車に乗ったり、水泳を覚えるときもうまくいった状況を再現するための全ての情報を筆記記録して、うまくできるようになるわけではない。もちろん、それらで習得したパターンマッチング的な認識は、属人的なもので他人には共有できない。そして、研究においても当の本人は、共有することのメリットを感じにくいのが普通である。結局、この属人的なパターンマッチング的な認識能力があることと、そもそもの属人的な認識を他人と共有することのメリットを意識できていないことがあるので、「実験及び分析を再現できるだけの情報記録」をすることの必要性を意識しにくいのである。

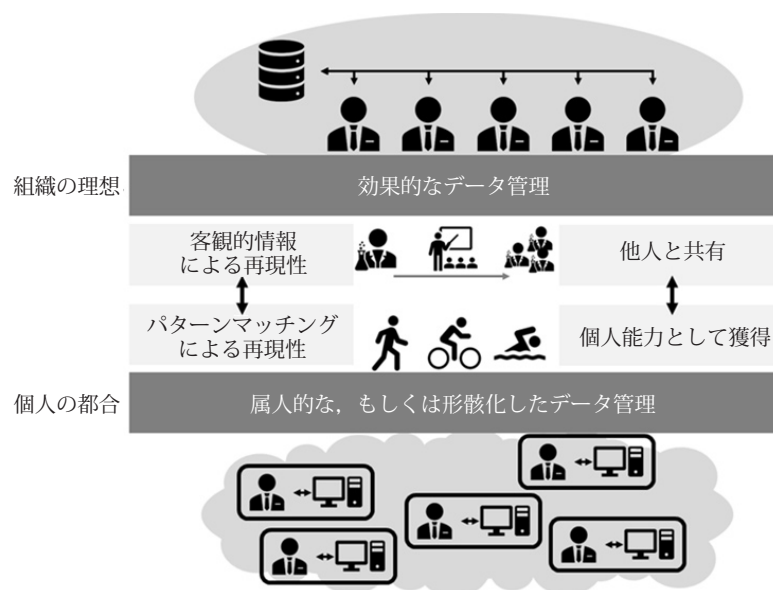


図3 データ管理に関する組織と個人の乖離の原因

「形骸化したデータ共有」状況が生まれる原因は、「属人的なデータ共有」状況が生み出される原因である「属人的なパターンマッチング的な認識能力」があるということと根本的な部分では同じではあるが、ここではもう少し掘り下げて、考察しておく。「形骸化」＝利用、活用されていない直接的原因は、大きく2つある。一つは、何かのデータを探したいと思った時に、どのように探せばいいのかわからない、そもそも探せるようにシステムが作られていないということである。これは、どのように利用するのかを十分調査、分析をしないでシステムを作ってしまうことが直接的な原因である。しかし、「どのように利用するのか」を調査、分析することは、実はそんなに簡単なことではない。実際、システムが存在しない状態で「どのように利用するのか」と聞いたところで、それは想像の中で「こう利用すると便利そうだ」と思っているだけで、実際にそのように利用して便利かどうかを保証した発言ではないのである。また、普段は「属人的なパターンマッチング的な認識能力」を使い（＝半分無意識で）データを探しているので、データの探し方は体系化、アルゴリズム化されているわけではなく、他人に説明できるようにはなっていない。その

ような状況下での「どのように利用するのか」の調査、分析がいかに困難かは想像できると思う。

もう一つは、蓄積されたデータが不完全で、「実験及び分析を再現できるだけの情報記録」となっていないことである。このような状態では、前項でも述べた「間違った結論が導かれたり、検討を進めたのちに始まりに立ち返って、再実験及び再分析を行なわざるを得なくなる」ことが起こってしまい、システムを信頼しなくなり、使われなくなってしまう。研究者同士の信頼関係が崩れないだけまだましだが、データ共有という側面では実質的には失敗である。なぜ、「実験及び分析を再現できるだけの情報記録」とならないかは、「属人的なデータ共有」状況が生み出される原因と同じ理由である。また、それを未然に検知できなかったのは「実験及び分析を再現できるだけの情報記録」とは何かを調査、分析できていなかったことが原因である。

4. 属人的データ共有状況を脱するための事前準備の前に行うべきこと

前項では、「属人的な、もしくは形骸化したデータ共有」状況が生み出される原因について説明した。本項では、データ共有システムを導入し、属人的共有状況を脱するために必要な事前準備の前に、どのようなことが必要かを説明する。

属人的、形骸化したデータ共有状態を脱する為には、データベース化／システム化以前に行うことが沢山ある。データベース化／システム化以前の単純なファイル共有レベルで、下記状況から脱却できている必要がある。

-1) 属人的データ共有状況からの脱却

-1.1 情報は記憶でなく、記録する。

⇒記憶は本人以外には見えないうえ、本人でさえ、時間が経つと忘却したり、改ざんされる。

-1.2 第三者が見て、認識に齟齬が出ないように項目名を定義し、合意を形成する。

⇒いくら記録をしても皆が同じ意味でとらえないなら、情報を共有していることにはならない。項目が増えてくると同じ項目に異なる項目名をつけたり、異なる項目を一つの項目として使ってしまうことが発生しうる。

0) 形骸化したデータ共有状況からの脱却

0.1 実験や結果分析の再現に十分な情報を特定・合意し、それを記録する

⇒誰しもが認識していない隠れパラメータは仕方ないが、そうでない限り、実験や結果分析の再現ができない記録はデータ共有の価値が大きく棄損し、実際、賢明な第三者は参照しなくなる。

0.2 利用、活用の観点から項目は適切に設定する

⇒例えば、ポリマ濃度／モノマ濃度で絞り込むことが多いのであれば、ポリマ濃度、モノマ濃度という項目があったとしてもポリマ濃度／モノマ濃度という項目がないと、全データを対象に演算を行うことが必要になる。それは、データ探査において、常に全データを取得する必要があるということであり、現実的でない。データベースも項目間の演算はできないので、データベース化したからといって解決する問題でもない。また、グラフ作成時の X,Y 系列に良く採用する項目に関しても、同じことが言える。つまり、データ共有をするからには、データ絞り込み、グラフ化でよく使うものは、あらかじめ項目化しておくべきということである。

これらは、データベース化、システム化という IT 観点で必要な事柄ではなく、データ共有のために必要な事柄である。したがって、それらを混同した議論や問題を起こさない為にも、これら対応が終わってからデータベース化、システム化の取り組みを行うことが肝要である。

5. R&D 部門におけるデータベース、システムは、魔法の箱ではない！

次にデータベース化、システム化のために直接的に必要な準備に関して、説明をしていこうと思うが、その前にデータベース、システムというものは、どういうものなのか？ R&D のデータベース、システムには何を期待していいのか？何は期待してはいけないのか？に関して、触れておく。結論を一言で書くと、「R&D のデータベース、システムは、魔法の箱ではなく、オルゴール！」である。データベース、システムというものを何でもできる魔法の箱のように思っている人が多い。それはある意味では正しいが、ある意味では間違っていると言える。

業務系のように既に業務内容が詳細に決まっており、それが単純な機能で長期間変化せず多くの人が使い続ける場合は、システムは魔法の箱のように作りあげることができる。つまり、項目や作業手順（データ絞り込み、データ処理）が明確になっていて、そのデータ絞り込み、データ処理が比較的簡単で、長期間変化しない場合は、そういうソフトウェアを開発し、システム内にあらかじめ組み込んでおけば、システムは魔法の箱である。ただし、非常に簡単な業務内容を実施できるだけでも、システム開発費は数千万から数億円になってしまうはずである。この業務内容が部門（経理、購買など）で会社に依存しない標準的なものでいいなら、数千万から数億円で開発したものを数百社に販売すれば、数十万円から数百万円の価格になる。そういう意味で、業務系システムは、自社専用にこだわらない限り、非常に安価な魔法の箱が実現可能なのである。

一方、R&D 部門では、業務系に比べ業務内容が詳細に決まっておらず、データ分析も一定ではなく人によって様々、つまり、属人的である。データ分析には既存の様々なツールを使う必要があり、ツールの機能も業務系のデータ処理とは比較にならないほど多様かつ複雑である。R&D 部門のデータ分析に使われるツールは、多様かつ複雑な機能を持つため業務系システムのようにシステム内に開発したものを組み込むことは、現実的でない。例えば、一般的なグラフ機能や画像処理等機能のすべてをシステムに組み込むなら、それらツールを製品として開発するのと同様以上の費用と時間がかかることになる。製品として販売されているツールの開発費は、どんなちっぽけなツールであっても、仕様作成、設計、開発、テスト、マニュアル作成を考えると開発費は、数千万円を下らない（数百本売れば、1本当たり十万円程度になる）。このことを考えれば、R&D 部門のシステムに一般的なグラフ機能や画像処理等機能のすべてを組み込むことが如何に現実的でないかは明白である。

したがって、R&D 部門のシステムでは、システム内にツールの機能を作りこむのではなく、システムから既存のツールを自動起動、自動処理する形で連携できるよう（オルゴールの外箱とドラムのように）にすることで、開発費の低減とメンテナンス性（ツールの変更や追加）の向上を図る必要がある。

ここまで、データベース、システムというものとツールというものを使い分けて説明をしているが、それらの違いが判るだろうか？誤解をしたまま読み進めないためにも、ここで本書内での使い分けに関して、説明をしておく。

ツール：研究者が欲するデータ処理機能を有するスタンドアローンソフトウェア

様々なデータ処理の機能を有するスタンドアローン（各 PC やサーバ等にインストールが必要な）ソフトウェアで、データの蓄積、共有機能やそれらを複数人で同時に利用するのに必要な運用機能が組み込まれていないもの

例）エクセル、グラフツール、機械学習ツールなど

データベース、システム：共有、検索、保全性等の裏方的機能を有するサーバソフトウェア

データの登録・更新、バックアップ、利用の開始・停止などの複数の人間が同時に同一のソフトウェア（データベース、システム）を利用しても問題が発生しないように設計されているもので、データの保全性や運用管理者と一般利用者などの役割を分けて機能利用ができる機構などの運用機能を備え、検索、処理に網羅性、均質性、再現性を保証することが大きな目的となっているもの

本来、データベース、システムで解決できることは、データ探査とデータ処理の自動化及び、それらに網羅性、均質性、

再現性という品質保証と作業者の省力化を付与することだけである。つまり、データベース、システムは魔法の箱ではなく、データを蓄積し、データをツールへ引き渡し、データ処理を自動化する部分はオルゴールの外箱で、どのような探査、処理をするのかはオルゴールのドラムである。どのような音（探査と処理）を奏でるかは、オルゴールを作る前に詳細に決められていなければならない。言い換えるとデータベース、システムがない状態で、既存ツールのみを使いデータ探査とデータ処理ができるところまでは達成できていなければデータベース、システムは意味がない、作れないということである。「どのような音を奏でるかは、オルゴールを作る前に詳細に決めておく」ということは、楽譜レベルに落とし込んでおく必要があるということで、実際、オルゴールのドラムに凸凹を刻み込む作業は楽譜がなければできない。「データベース、システムない状態で、データ探査とデータ処理ができるところまで」というのは、楽譜ではなく、データ探査とデータ処理を誰でも読めば実施できるレベルの手順書にまとめられているということと同義である。そして、ドラムに凸凹を刻み込む作業は、手順書をシステムに組み込む作業に対応するのである。

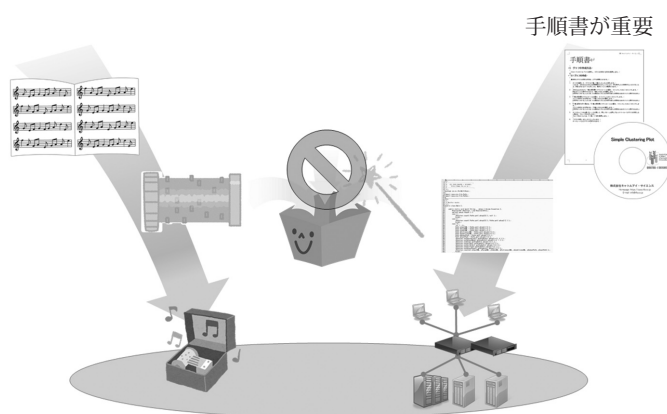


図4 データベースやシステムは魔法の箱ではなくオルゴール！

したがって、データベース化、システム化で課題を解決するためには、データベース、システムがない状態（ツールは使ってよい）で、研究者でない第三者が手動処理で確実に実施できる手順が確立されている必要がある。実際、この手順書がなければ、システム仕様が決められないし、システムが意図通りに作られたかどうか確認できない。つまり、この手順書を書き下すことがデータ共有・利活用の最初の一步ということである。

6. データベース、システムの最大の利点とは！

ここで、そもそもなぜ、データ共有・利活用をするためには、データベース化、システム化をすべきなのかを考えておく。データ共有・利活用をするためには、実験、分析の再現ができるレベルの記録すべき項目を明確化し、それら項目を使って、どのようにデータを探査し、データ処理をするのかを第三者が実施できるレベルの手順書にまとめおく必要がある。実は、これだけでもデータ共有・利活用は可能である。そういう意味でいうと、データベース化、システム化は本質でないとえばその通りである。しかし、オルゴールの例を思い返してみると、手順書止まりということは楽譜止まりで、データベース化、システム化をするとオルゴールになるのである。何が違うかというと、オルゴールは、ドラムを作った人の意図通りに常に同じ音楽を何度でも奏でるが、楽譜は奏者によるばらつきが生まれ、それこそ熟達した奏者でないとメトロノームがないと一定テンポで演奏し続けることさえ難しい。オルゴールの価値は、まさしく音楽を均質に何度でも再現することができることである。つまり、データベース、システムは、オルゴールと同じようにデータ探査、データ処理に網羅性、均質性、再現性を付与するためのもので、まさしく、データベース、システムの価値はここに見出すべきである。データベース化、システム化は、これ以外にも作業負担軽減、作業時間短縮というメリットもある。これは、手動処理で網羅性、均質性、再現性を達成できるような業務では主たるメリットであるが、R&D 部門に限って言うと、手動処理で網羅性、均質性、再現性を達成できることはまずないため、

作業負担軽減、作業時間短縮はあくまで副次的なメリットであり、データベース化、システム化の目的は、データ探索、データ処理に網羅性、均質性、再現性をもたらすことと考えるべきである。そう考えられない場合は、「データ探索、データ処理に網羅性、均質性、再現性」がないと、どのようなことが起こるかを再度思い返してほしい。以下で、網羅性、均質性、再現性に関して、少し解説を加えておく。

網羅性: 手動で全データを対象に探索、処理を行うことが現実的ではないが、システムだと全データを網羅した探索、処理が可能である。

例) 熱伝導度が〇〇以上の材料を探したい場合に、記憶からいつぐらの実験にあったはずと目星をつけて、いくつかのデータを確認する。しかし、目星が間違っていることもあるし、そもそも目星が「熱伝導度が〇〇以上の全ての材料」につけられるわけではない。このような網羅的でない抽出データは傾向が偏っている可能性もあり、それを分析した結論は、信頼性を大きく棄損している。

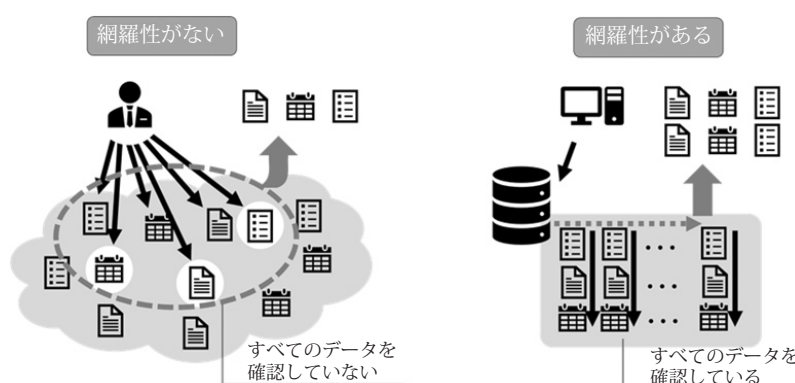


図5 データベースの利点の網羅性とは

均質性: 手動で探索、処理すると意識バイアスがかかったり、初期と終期での同一性が維持できないが、システムだと完全に均質な探索、処理が可能である。

例) ガラス転移点が 212 度以上の材料を探している場合に、いくつか見つけた後、あまりにも該当データが少なかったため、210 度以上のものも含めるようにした。ただ、最初の方から探索をやり直していないので、いくつかのデータは取りこぼしている可能性がある。また、明らかに他の特性が合致しないデータは、212 度以上でも抽出しないことにしたが、明示的なルールで一貫しているかといわれるとグレーである。このような均質でない抽出データを分析した結論は、信頼性を大きく棄損している。

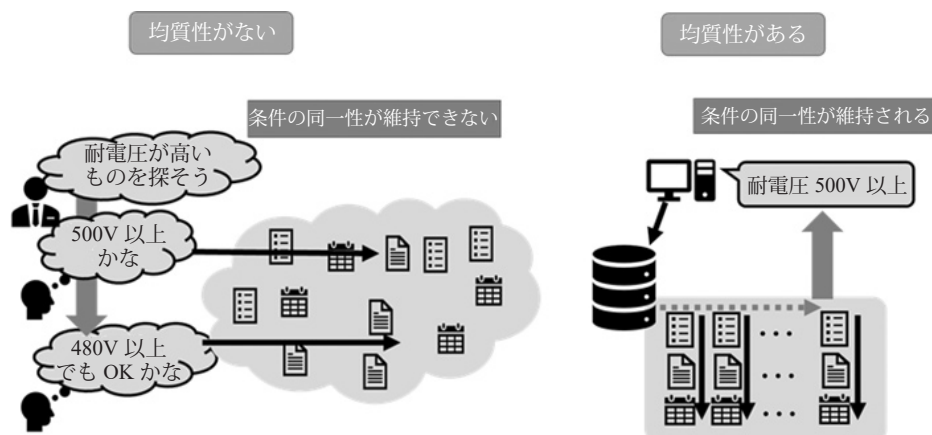


図6 データベースの利点の均質性とは

再現性：半年前のデータ探査や処理は、手動ではまず再現できないが、システムであれば、データ探査や処理の再現が可能である。

例) 手動処理では、上記で述べたように網羅性も均質性もほとんどの場合で担保されていないので、当然、再現性も期待できない。こういう再現性のない抽出データを分析した結論は、信頼性を大きく棄損している。

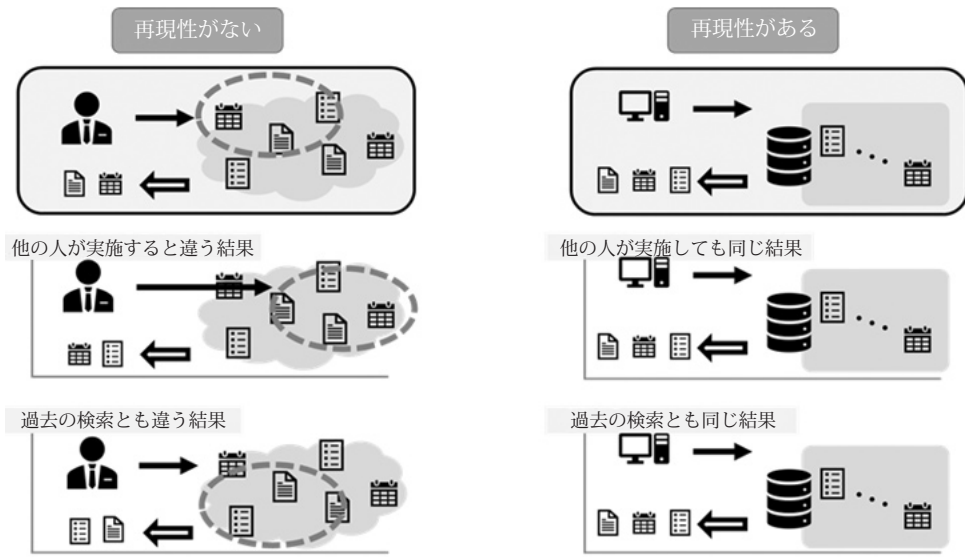


図7 データベースの利点の再現性とは

7. 属人的データ共有状況を脱するための事前準備（手順書作成編）

話が脱線しすぎたので、話を戻す。データ共有のために必要な準備（実験や分析を第三者が再現するのに十分な情報及び利用、活用観点を考慮した項目のリストアップと合意形成）が終わった後、にデータベース化、システム化のために直接的に必要な準備を行う。「システムはオルゴールのようなものだ」という話を思い出してもらおうと、何をしないといけないかは自ずとわかってくる。そう！楽譜が必要なのである。つまり、「データ探査とデータ処理を誰が読んでも実施できるレベルの手順書」を作ることが最初の課題である。

「データ探査とデータ処理を誰が読んでも実施できるレベルの手順書」を作るとは、非常に簡単なようで、なかなか難しい。データ探査とデータ処理自体は、データベースやシステムがない時からすでに行っているはずである。ただ、それを手順書としてまとめている人は、研究者の中では非常に稀な存在で、ほとんどの研究者は経験という記憶に頼って行っているはずである。経験を的確な文書に起こすことが難しいことは、自転車の補助輪をどのようにして外せるようになったか？、どのようにして平泳ぎができるようになったか？、どのようにして、日本語を話せるようになったか？を考えてみればわかるはずである。ただし、データ探査とデータ処理に関しては、これができない限り、システム化はできない。実際、補助輪外しや水泳や日本語取得に比べると格段に簡単な課題なので、覚悟を決めて取り組み、慣れてしまえば、この程度の経験を手順書に落とすのは簡単である。研究の中核的な思い付き（この材料を配合してみれば良さそうだ、ここの攪拌は泡立てないように注意しながら）は、どのようにして思いつくかを手順書に落とすことは、非常に難しい（とはいえ、本当はグループリーダからすればそれをしたいのだと思うが・・・）はずで、それを要求されているわけではないので、データ探査とデータ処理の手順化ぐらいは頑張ってもらいたい。

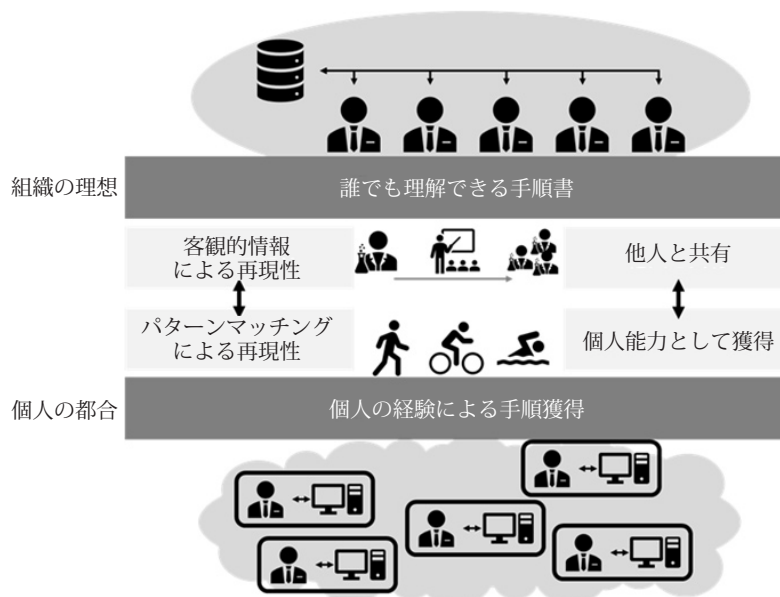


図8 他の人でも実施できる手順書作成の難しさ

手順書を書いていくといくつかの課題にぶち当たるはずである。一つ目は、手順が確定していないことが発覚するということである。「実験や分析を第三者が再現するのに十分な情報及び利用、活用観点を考慮した項目のリストアップと合意形成」は終わっているが、実際にまだデータ共有・利活用を行っていないかたりしていると実験を記録したファイルの保存場所がまだ、決められていない（属人的）などの問題が浮上していない場合がある。当然、このような課題が出てくるたびに合意形成をし、改善していくことが必要である。想像だけで大丈夫と思っても、案外抜けがたたくさんあるもので、合意形成が終わった後で、実際にそれに沿って、データ共有・利活用を実践しておくことが重要ということである。「これをシステムで改善したいんだ」という話をよく聞くと、システムでも実験データファイルを適当な場所に置かれるとそれを見つけて保存することはできないし、そもそもシステム化するとある一定のルールに従って、データを登録してもらうことになる。つまり、システムがない段階で、一定のルールに従って、実験データファイルを置けないのであれば、システム化したってデータは登録されないだけである。「これをシステムで改善したいんだ」と思った場合、「システムは魔法の箱ではない」ということを思い出して、手順書作成を進めてほしい。

二つ目は、「多種多様なすべてのデータ探索と処理の手順書なんて、書けるわけがない」というものである。これは、ある意味では正しいが、ある意味では間違っていると言える。R&Dでは、経理や財務のように完全に決まりきったデータ探索と処理はほとんどなく、それを逐一挙げていくのは、現実的ではない。また、今日、新しいデータ探索と処理が発生するかもしれない。しかし、システム化をするときには、それでは困るのである。どうデータ探索をして、どうデータ処理をするのか？ 詳細な情報なしでそれができそうな魔法の箱はないのである。「だって、エクセルやグラフソフトや実験計測装置についているソフトは、詳細な情報なしでデータ処理ができるように作られているじゃないか？」、「googleも我々の詳細な情報なしで世界中の情報を探索できるじゃないか？」と言った感じで、そんなことはないはずだという人もいるかもしれない。しかし、それはどんなことでもできるようコストをかけて、様々な機能を組み込んであるか、深く洞察して見ると、実は単純なことしかできなかったことが見えてくる。エクセルが長い期間をかけて現行の機能を実現するために費やした開発費は数百億円はくだらないだろうし、googleの検索では、項目名を指定した検索（圧力が〇〇以上など）は、そもそも目指していないのである。また、なんでもできるソフトウェアを使って、自分たちが利用しようとしていることを実現しようすると結構難しいもので、高機能なエクセルなどは、よく知っている人やネットで、使い方を調べないといけないことがよくあるはずである。実はもっと汎用的なソフトウェアが存在するそれは、プログラミング言語である。エクセルもワードもパワーポイントもプログラミング言語で作られている。巷を賑わせている機械学習の多くもpythonというプログラミング言語を使っている。プログラ

ミング言語も歴としたソフトウェアで、ある意味汎用ソフトウェアの極みである。何を言いたいかというと、様々なことができ、汎用的になればなるほど、使いこなすためのハードルがあがり、実際使う力があつたとしても多くの手数がかかるということである。そういう意味でも魔法の箱は存在しない（あつたとしたら使いこなすためのハードルが高く、使うときの呪文がやたら長くなるので、便利ではない）のである。世の中で便利だなというものは、ある一定の場面に特化していて、その場面で発生するあらゆることをできるだけ簡単に実現できるように設計されているのである。つまり、「すべてのデータ探索と処理」の場面で発生するあらゆることがわからないとそれをオルゴールのドラムに刻み込めないのである。しかし、「その場面で発生するあらゆること」を本当に手順化しているかというそれは少し違う。スマートフォンの音楽アプリで音楽を聴く時に、「アプリを起動して、〇〇という音楽を選択して・・・」というときに、〇〇の楽曲名ごとに手順書を書いていたら大変であるし、実際、アプリの開発者はそんなことはしていない。楽曲部分は、変数化して手順を書くのである。

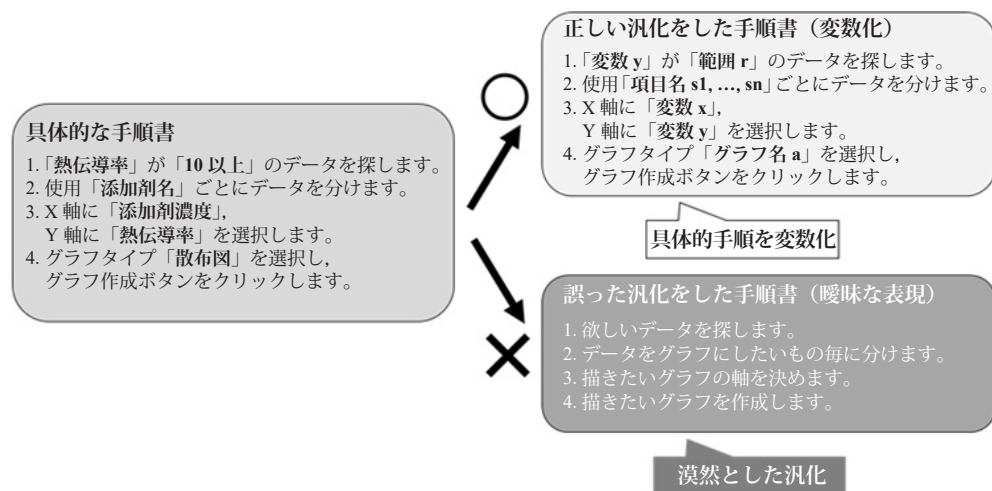


図9 手順書の正しい汎化方法

ただし、一度も具体的に楽曲を探さないで手順書を書くことと失敗する。楽曲名がわからないことも多いし、同じような楽曲名や全く同じ楽曲名があるとき、「〇〇という音楽を選択して」の前に、「最近の」とか「△△が歌っている」という絞り込みを入れないといけないはずである。この辺りはいろんなパターンがあると思うので、そのパターンが出尽くすように色々具体例で手順書を書き、その手順通り実施をしてみて、その後、具体的な部分を〇〇、△△という変数に置き換える（汎化する）のである。また、汎化の時に正確性を犠牲にして、曖昧になってしまうことがよくあるが、実際には数学の変数式のように一般化されているが厳密性は一切劣化していないものでないといけない。簡単に「具体的な部分を変数に置き換える」と書いているが、この変数名が具体的なものをすべての確に言い表す単語でないと、結局自分が行いたいことの手順を知ろうとして手順書を見ても変数名がいい加減だと、自分が今しようとしていることの何をどこに置き換えればいいのかわからない状態になる。変数名は重要なので、頭を悩ませるところであるが、しっかりと考え抜く必要がある。実際には、「様々あるすべてのデータ探索と処理の手順書」も本当にすべてを列挙するのではなく、できるだけ多くのパターンに関して具体的な値で手順書を作成し、手順書通りで意図通りのことができるかを確認してから、具体的な部分を変数に置き換えて汎化した手順書を作る必要がある。そうすると手順書はそんなに多くの数にはならないし、稀な作業に関しては、今回のシステムでの利用対象外として、手順書を書くのをやめてもよい。どちらにしても「様々あるすべてのデータ探索と処理」のパターンと頻度は、把握しておく必要はある。そして、これはソフトウェア会社にはできず、実際の研究者しかできないことである。

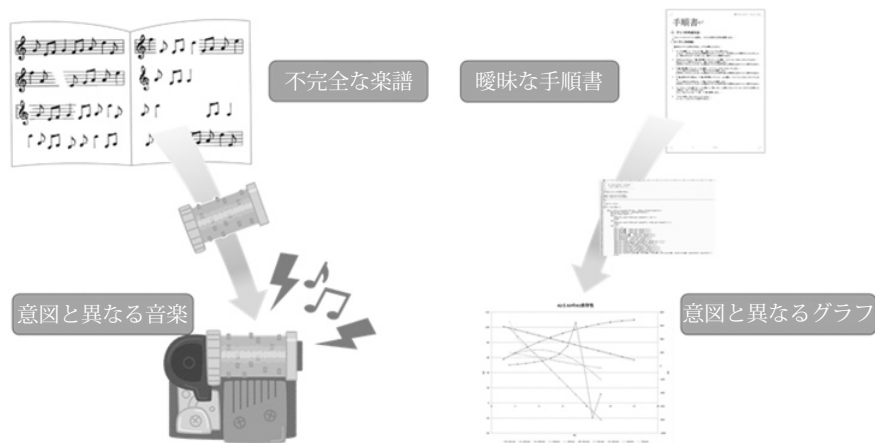


図 10 曖昧な手順書は意図とは異なる結果をもたらす

手順書づくりの中で、取り組んだほうが良いことがもう一つある。データ探査やデータ処理の手順が明確になったら、「なぜ、そのデータ探査をするのか?」、「なぜ、そのデータ処理をするのか?」を複数の研究者で意見交換をし、合意を形成してみることである。たぶん、漠然とした答えはすぐ出るだろうが、新人や素人にその理由がはっきりとわかるというレベルで、明確な答えが出ることは少ないはずである。もちろん、専門的な背景の有無により、わかるわからないということはあると思うが、それ以上に曖昧になってしまっているはずである。手順の話と同じで、人間の頭の中で分かっていることも他人にはそれを伝えられないし、わかっていると思っている人同士でも思っていることが同じかどうかは確かめられないのである。そして、「なぜ」が明確になっていないということは、目的もはっきりしていないということで、そのように目的が曖昧なままの状態で作業をすると、結論もおかしくなるのは皆様よくお分かりの通りである。また、「なぜ、そのような方法でデータ探査をするのか?」、「なぜ、そのような方法でデータ処理をするのか?」も同じように複数の研究者で意見交換をし、合意を形成することは、非常に重要ということである。これらの「なぜ」の真の目的を明確化するためには、障害の原因分析のよく使われている「なぜなぜ分析」という方法が便利である。5 回程度「なぜ」を繰り返すことで、深堀ができ、真因に近づけるはずで、「なぜなぜ分析」自身の方法は、Web でもいろいろな解説や注意点が書かれているので、漠然とではなく、必ずそちらを参照して実施してほしい。真の目的が明確になれば、真の目的からデータ探査やデータ処理方法を見つめ直し、最適でないことが分かれば手順書を改善することが必要である。

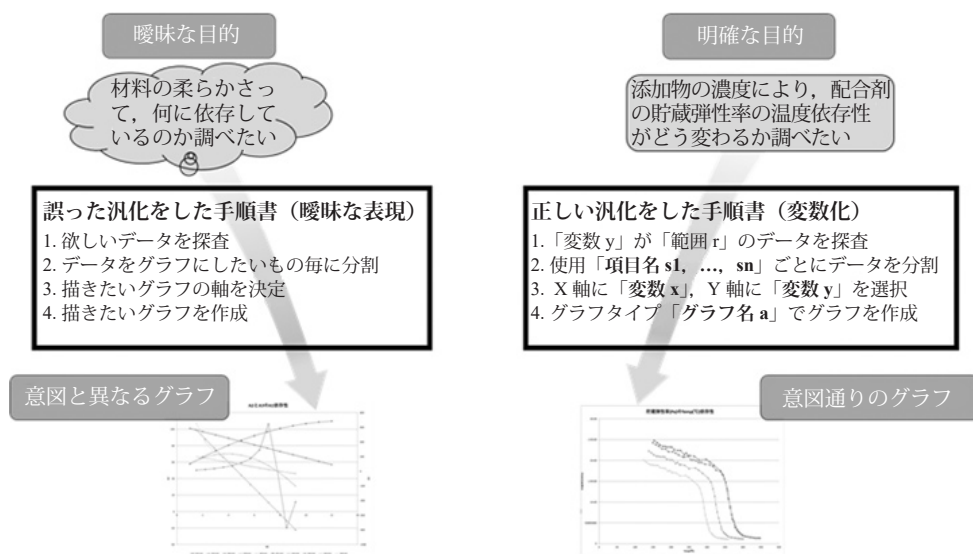


図 11 目的を明確にしないと手順も結果も意図通りにはならない

手順書ができれば、実際にその手順書が汎用的に利用可能かを検証する意味を含め、その手順が発生すると手順書の各工程にかかる時間を典型的な数パターンで測定し、記録する。時間を記録するために手順書通りに実施するのではなく、それこそ新人などをお願いして、日常業務を遂行するために手順書を使い、ついでに時間を記録してもらうのである。それなりの日数をかけることで、手順書の問題も明らかになってくるはずである。頻度に関しては、ある程度は実測できるだろうが、限られた時間ではどうしても偏ってしまうので、全員で意見を出し合い、合意形成をするしかない。ここで、手順の問題を明確化しておかないとシステム化してからでは、原因がどこにあるのかを特定する作業や手戻りが大きすぎる。急がば回れである。

8. 属人的データ共有状況を脱するための事前準備（システム化対象特定編）

データ共有・利活用のシステム化及びシステム運用には必ず負担がかかる。したがって、闇雲にいろいろな作業をシステム化の対象と考えるのではなく、作業に品質（網羅性、均質性、再現性）が重要なもの及び作業の発生頻度と作業にかかるネット時間（研究者が実質的に作業をしている時間で、待ち時間や他の作業をしている時間を除く）の大きいものを序列化し、その上位の作業からシステム化対象にあてていくべきである。実際、この序列は現業務の問題の深刻度とも対応するはずで、もし、現状深刻な問題を感じている業務が上記序列で上位にあがってきていない場合は、頻度や作業時間の算定が間違っているか？主観的な思い込みのバイアスがかかっているか？のいずれかである。複数の人と意見交換することや再計測をすることで、その認識ずれは解消しておく必要がある。

システム化対象が決まったら、次はどのようにシステム化を行うかを検討をしていく。この段階になって、初めてソフトウェアの観点を加味して考えることになり、システム開発関係者も参加して意見交換を交わすことになる。この時、注意すべき点がいくつかあるので、ここで触れておく。あくまでシステム化の主目的は、データ共有・利活用に対して、網羅性、均質性、再現性という品質を担保するためのものである。もちろん、作業の発生頻度×ネット時間が大きいものもシステム化の対象にはなるが、安易に作業負担をソフトウェアで軽減させよう考えてはいけない。通常「網羅性、均質性、再現性」を維持した手動作業より、システム化した場合の作業のネット時間が大きくなることは考えにくいので、まずは「より使いやすく」、「より便利に」ということは考えずに、手順が確実に実行できるかどうかだけに焦点を絞ってシステム化を考えるべきである。つまり、システム化に便乗して「より楽に」ということを考えてはいけないということである。業務系システムでは、手順が明確になっており、手順の実行頻度も安定的に固定化されている。また、業務系システムの変更は簡単に実施できないので、1回のシステム導入や更新でできるだけ使用者が「より楽に」作業遂行できるように配慮する。しかしながら、R&Dでのシステムでは、R&Dの宿命上、多くの手順変更や新たな手順が発生し続ける。つまり、できるだけ迅速に手順追加や手順変更に従従できることが最優先課題で、「より楽に」、「より便利に」ということは次点以下の課題なのである。実際、「より楽に」、「より便利に」を考慮しだすと結局、運用、メンテナンスが追い付かなくなり、本末転倒な状態になる。手動手順と同等のことが同等時間以下の作業で達成できれば、もれなく、網羅性、均質性、再現性という品質がついてくるので、これを大きな価値とする考えを持つべきである。実際、R&D部門で、データ共有・利活用がシステムでうまくいっているところはすべて、このような方針が徹底された上で、開発、運用を行っている。

実際、「より楽に」、「より便利に」は、客観的でない（主観）欲求であることがほとんどで、客観的負担対効果を蔑ろにする考えであり、システム開発、利用に行き詰まりを生む元凶である。「より楽に」「より便利に」という要求があった場合、あくまで、手動で共有・利活用をした時と比べて、どれだけ省力化できるかどうかという客観的指標で考えるようにする。そして、この客観的省力化量がシステム化の優先序列であり、省力化量の小さいものは、客観的負担対効果は小さく、取り扱うべきでないと判断をすべきである。もちろん、「より楽に」、「より便利に」の部分が、客観的省力化量が大きい部分があるのであれば、それは採用されてしかるべきである。

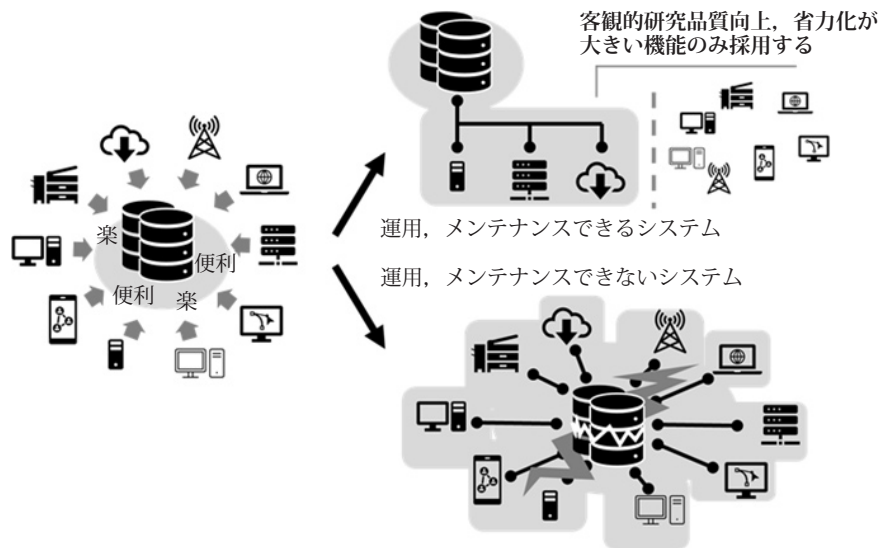


図 12 便利と楽を追及すると運用、メンテナンスできないシステムになってしまう

システム化に便乗して、今まで行っていないデータ探索、データ処理をこの機にシステムに実装してしまいたいという要望が挙がるがよくある。しかし、これは絶対に行ってはならないことの一つである。以下、なぜこのようなことは行ってはいけないのかを説明しておく。

1) 第三者が手動処理で確実に実施できる手順書がない

システムは、オルゴールの外箱であり、楽譜があって、それをドラムに刻み込まなければシステムは完成しないことは上述している。手順書がないということは、楽譜がないということである。楽譜なしに音楽をドラムに刻み込むのは不可能であり、また、即興で楽譜を書いたとしても天才でない限り、まともな音楽になっていることは奇跡である。実際、すでに手動実施していることを手順書に書き下すことでさえ、悪戦苦闘している状況を考えると、如何に現実的でないか理解できると思う。当然、そのような楽譜や音楽は、オルゴールに組み込むべきではない。

2) その作業が他の作業よりシステム化するべきか客観的に評価できない

即興で実際に実施可能な手順書が書けたとしても、「その手順通り実施した結果が価値の高いものか？」や「その実施頻度がどの程度か？」は、判断がつかないということである。つまり、そもそもシステム化対象を決める序列の「作業に品質（網羅性、均質性、再現性）の重要性」、「作業の発生頻度×ネット時間の大きさ」の正確な値が算出できず、他のものと客観的に比較ができないのである。

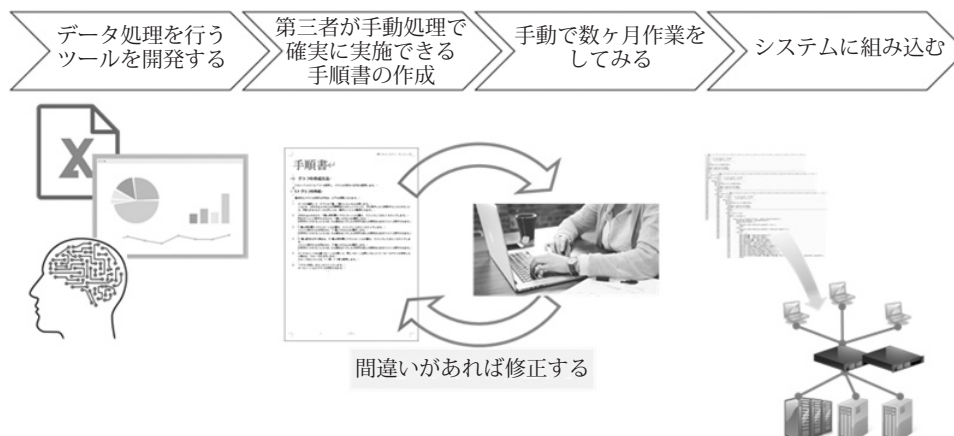


図 13 手動で行えていないことをシステムに組み込むまでの手順

「今まで行っていないデータ探査、データ処理をこの機にシステムに実装してしまいたいという要望」の中で、ツール自体もまだ存在していない場合もある。当然このような作業のシステム化は、上記理由より行うべきではないが、「現在、ツールが存在していない」状態で、システム化を行ってはいけないもう一つの重要な理由を以下で説明しておく。

3) ツールとシステムの境界が曖昧になる

R&D 部門では項目追加、変更だけでなく、データ処理に関しても追加、変更が頻出する。これが業務系と異なり、システム化を難しくしている原因である。このような R&D 部門でもシステム化を可能にするためには、ツールとシステムの分離が重要で、システムからツールを呼び出す形のシステムでなければならない。そのようにすることで、システムなしにツールを使い、手順を確立することが可能になる。しかし、ツールとシステムを同時に開発すると、手動手順の確立ができなくなってしまう。また、同時開発はツールとシステムの境界を曖昧にしまい、その結果、ツール部のみの改良やシステムを改良したときにツール部に問題が生じないかの確認、検証が難しくなる。つまり、ツールやシステムのバージョンアップ時にシステム自体の長期の停止が必要になったりするなど、メンテナンス性が悪くなるのである。

4) ツール部、システム部及び手順の問題の切り分けができない

ツールとシステムを同時に開発すると、何か問題があったときに、手順に問題があったのか？、ツール仕様に問題があったのか？、システムに問題があったのか？の切り分けが非常に難しくなる。そもそも、システムを使わない状態での正解が示されないとその判別自体が原理的にはできないということになる。システムの問題であることを排除するためには、手動でツールのみを使った手順の確立が必須である。もちろん、ツールの問題であることを排除するためには、ツールを使わない手動手順の確立が重要である。したがって、ツールを自作する場合には、他の複数の商用ツールなどでも同じことができる手順を確立しておくべきである。また、システム開発会社にすべてを丸投げしてしまう場合は、丸投げ体質が染みつき、もし、ツール部やシステム部に問題があっても、気づかず間違った分析をしたままになる可能性が高くなる。実際、正確な仕様書と手動手順書がないとシステム会社も何が正しいのかを把握していないことになるので、わからないところに丸投げをするという非常に危機的な状態に陥る。

「それでは、将来展望など、現時点でできないことを実現するためにはどうすればいいのか？」という質問がよくある。すでに答えは上述しているが、以下でまとめておく。

1) データ処理を行うツールを開発する

データ処理を行うツールが存在しないのであれば、まず、手動手順での実施を可能にするためのツール作成を行う。ツールの動作を確認する為のサンプルデータとその結果も整えておくこと。実際、ツールに BUG がないことを保証するのは、相当大変だが、システムに組み込む前に十分確認しておかないと、皆がツールには問題がないとして使いだすので、十分注意が必要である。もし、ツールをリリースした1年後などに問題が見つかった場合は、その連絡および関係部署への影響範囲の確認などが発生し、結構大きな問題になる。

2) 第三者が手動処理で確実に実施できる手順書の作成

ツールを使って、手動で実施可能な手順を確立し、手順書を作成する。また、この過程で、データの項目名や項目値の単位がばらばらだとか、数値の項目値に数値以外の値が入っている(> 1, ~ 10, "とか、目視では意味が分かるが、システムには意味判別ができない記述)ものは、適切な数値に修正をしたり、新たな項目(下限値, 上限値, 信頼幅など)を作り、その値に変更するなどの作業をする。この作業は一般的にデータクレンジングと呼ぶ。

9. 属人的データ共有状況を脱するためのデータ共有システム導入に必要な要件

前項では、「属人的データ共有状況を脱するための事前準備」に関して、説明した。本項では、R&D 部門の属人的共有状況を脱するために必要なデータ共有システムのソフトウェア要件を説明する。

- 1) 項目名の単語ブレや他データとの紐づけ時のミッシングリンクが発生しないように、データ登録や検索部分では項目名や紐づけはデータベースと連動したドロップダウンリストから選択できる機能を有すること。また、DB と不整合があったときには警告メッセージなどが出て、登録前に抑止できる機能を有すること。
- 2) 外注をしなくても研究者自身で項目名の変更、追加や他データとの紐づけ変更、追加の対応が可能な機能を有すること。
- 3) データ登録の入力部分は、外注をしなくても研究者自身でカスタマイズができる機能を有すること。具体的には、データ登録の入力部分は研究者が日頃から使用しているエクセルシートであることが望ましく、実際に登録するデータはその入力シートの登録したい部分を参照する形になっていることが望ましい。こうすることで、調液表などのように複雑な入力、確認が必要な場合でも研究者自身でエクセルシート内で演算をしたり、項目値についても選択肢を設定できたり、必須項目を設定したり、入力時の確認欄（合計値が 100% になるなど）を設けたりすることが可能となる。つまり R&D 部門の多様性、変化性に追従することが可能になる。
- 4) 検索結果のデータを他のサーバ、PC の既存アプリケーションに処理を受け渡すことができる機能を有すること。具体的には、Linux や Windows にそれぞれの利用者アカウントでログインし、既存アプリケーションをコマンド実行し、結果ファイル等をシステムに自動回収し、同時にデータベース化を行う機能を作りこむのではなく、システム基盤機能として有していること。

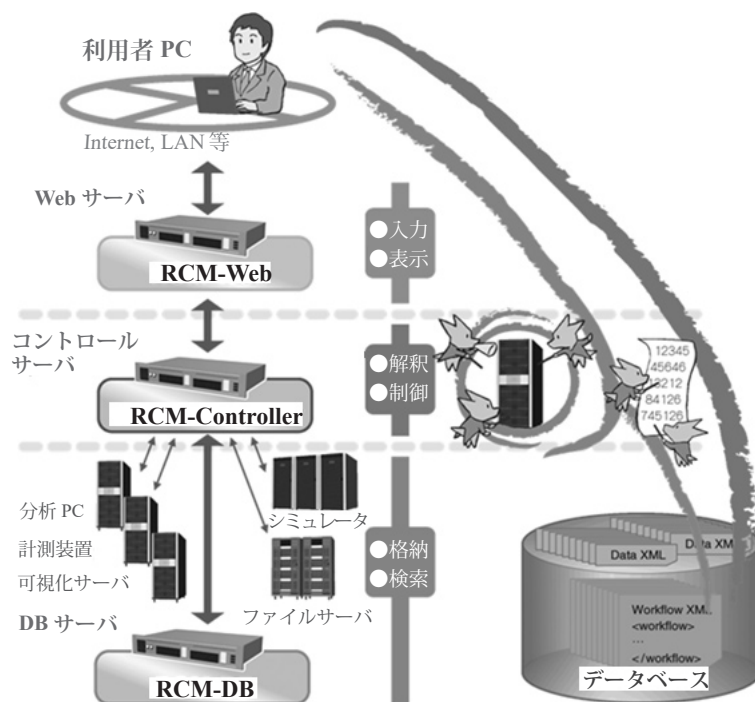


図 14 R&D の変化に追従するために必要なシステム

- 5) 機能変更・追加が頻繁に必要となるので、機能拡張が容易となる機構を有していること。具体的には、スキーマレスなデータベースを持ち、システムから DB やツールを起動し、様々な処理をする部分を追加変更するにあたり、システムのリビルドが不要な機構を有すること。
- 6) 一つのシステムハードウェア内に複数の独立したサブシステムをくみ上げることができ、アクセス制御を行うことで、サブシステム間の情報のやり取りも可能な仕組みを非仮想環境上で実現する機構を有すること。サブシステムごとに仮想環境にすると、独立性はより高くなるが、メモリや CPU や HDD のリソースまで完全独立になり、また、OS 部がサブシステム分だけ重複し、非常に非効率なシステムとなってしまう、要求ハードウェアが数倍以上ハイスペックで、一桁以上高価格になる。
- 7) R&D 部門用システムでは、機能変更・追加が頻繁なので、システム無停止で、サブシステムの追加や変更が可能な機構を有すること。他グループのシステム追加や機能拡張のために他のグループのサブシステムが運用停止することは、受け入れられないことが多い。
- 8) アクセス制御がプログラムレベルではなく、DBMS レベルで制御されていること。R&D 部門用システムでは、様々なカスタマイズ開発が行われていくはずで、プログラムレベルで、アクセス制御を行う仕組み（データベースにアクセス制御のための情報を登録しておき、それをプログラムで判断して、アクセス制御に合致する結果だけを返すようにする）になっていると、「アクセス制御に合致する結果だけを返す」というプログラム部に不具合があると簡単に、情報漏洩が起きてしまう。カスタマイズ開発で変更できないシステムの基盤部である DB 部でアクセス制御は判断されるような機構を有している必要がある。

10. データ共有システム導入時の落とし穴とそれを防ぐ方策

前項では、「R&D 部門の属人的共有状況を脱するために必要なデータ共有システムのソフトウェア要件」に関して、説明した。本項では、R&D 部門におけるデータ共有システム導入時の落とし穴とそれを防ぐ方策に関して説明する。

- 1) 項目名や項目間の関係性を明確化する段階で議論が紛糾し、纏まらない

⇒まず、目的を再確認し、「項目名や項目間の関係性を明確にすること」は、目的のためになぜ必要なのかを皆で共有することが必要である。その上で、この議論が収束しない場合は、本課題を進めている人が適切かどうか再検討する必要がある。この手の合意形成を進めることができる人は、複雑な利害関係の絡んだプロジェクトを利害関係者間の落としどころを探りながら進めていける技量のある人である。単に、紛糾している状態を丸く収めるだけでは、問題を先送りしているだけであり、ガチンコ状態で、合意形成を取り付ける技量が必要である。そもそも、皆がガチンコ状態になっていない状態であれば、ガチンコ状態に持つていくこと自身も大きな仕事である。たぶん、グループリーダなどが行うことが一番理想的であるが、リーダー以外にもこういうことが得意な人物がいれば、リーダーが背後に立ち、その人物に任せるのも一案である。社内の適切な人員が見当たらない場合は、こういうことの経験のあるコンサルタントをファシリテータとして、採用、外注することも考えるべきである。

- 2) 項目名などを一元化するための管理者が必要だが、それを決められない

⇒「項目名や項目間の関係性を明確にすること」ができる人なら、この管理者も担うことができる。システムの管理者としての実務の管理者は、決められた作業をしておけばいいだけである。したがって、「項目名などを一元化することを尽力する人」は、1) と同じ人が担い、システムの管理者としての作業実務部分は、操作ミスを起こさない慎重な人に担ってもらえばよい（もちろん、どんな慎重な人でもヒューマンエラーは起こ

り得るのでダブルチェック体制は必要)。

3) システムが存在しない状態で、どのような機能を有するべきかを考えることは難しい

⇒ユーザーが使いたい機能だけでなく、データ保全を意識した運用に必要な機能は、研究者にはどんな機能が必要なのか？また、どんな仕様であるべきなのか？の判断は困難である。R&Dにおける運用実績が豊富なパッケージ（機能を作らなくてよい）を採用するか、運用に必要な機能を研究者と合意形成を図ることができる R&D 部門のシステム開発、構築のコンサルティング経験のあるコンサルタントを雇うべきである。実際、こういうことをせずシステムを構築してしまうとシステムというより、ツールの（ツールは、その時のデータでその時に結果を返せばいいだけで、項目名が揃っているか？とか、同じことを他者ができるかという観点が極めて希薄）になってしまい。複数人の同時使用やバージョンアップ、データの意図しない削除など様々な問題を抱えたまま、その問題を認識しないまま運用してしまうことが非常に多い。また、そういうツールのシステムは障害が認知された場合でも、何をして、どういう問題が起こったかなどの記録が残っていないことが多く、原因究明ができず、問題を放置した状態になることが多い。そうすると、信頼性を失ってしまい、次第に使われなくなってしまう。

4) 何をどのように探し、どのようにデータ処理しているのかの現状調査、分析が纏められない

⇒これは、地道に具体的な作業を観察し、「何をどのように探し、データ処理しているのか」を調べ、記録する力とそれを汎化（パターン化）していく力が必要である。前者は、地道な作業ができる人、他の人に頼み事（実際に作業をしてもらうことを依頼し、横から観測して、何をしているかを記録する）するのが得意な人が適しており、後者はロジカルシンキング能力が高い人が適している。前者は、どんなグループにも一人ぐらいいいような気がするが、後者は理系研究者の中でも適する人は少なく、実は文系の企画担当や法律家やネゴシエータに適する人が多い。ただし、文系の企画担当や法律家やネゴシエータにとって、研究開発の基礎部分は単語一つをとっても今まで扱っていないものが多く、力が発揮できるまで、それなりの時間を要する。社内の適切な人員が見当たらない場合は、こういうことの経験のあるコンサルタントをファシリテータとして、採用、外注することも考えるべきである。

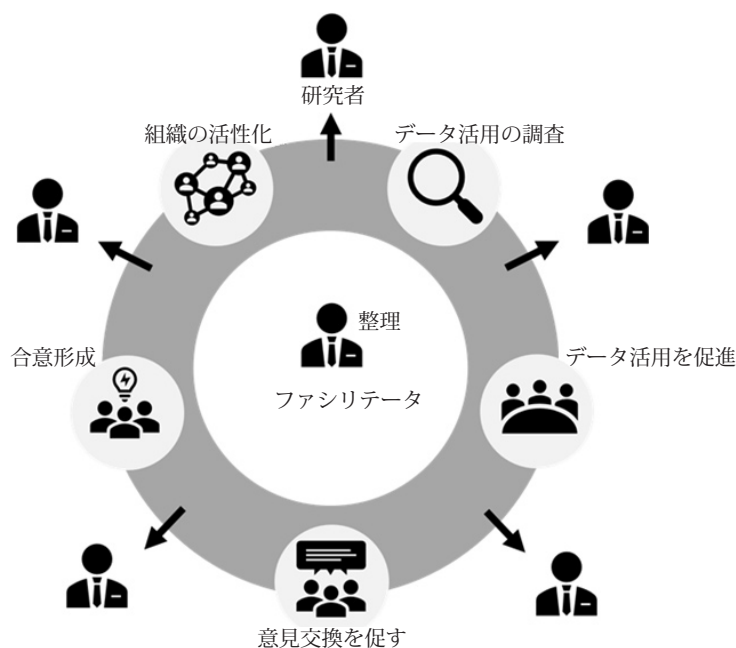


図 15 ファシリテータによる現状のデータ利用の実態や問題を把握、改善

ここまでの説明で、何度となくコンサルタントやファシリテータという言葉が出てきている。R&D 部門のデータ共有システムを導入、運用する際にコンサルタントやファシリテータに必要とされる力を以下にまとめておく。

コンサルタントやファシリテータに必要な能力は、研究者との深いコミュニケーション力とロジカルシンキング力。実際の研究者が行うしか仕方がない以下の課題の部分を支援し、完遂させる能力である。

- 1) 属人的データ共有状況からの脱却（情報記録，項目名定義）
- 0) 形骸化したデータ共有状況からの脱却（十分な情報，適切な項目）
 - 1) 第三者が手動処理で確実に実施できる手順書の作成と汎化
 - 2) 手順書作成時の問題点の列挙と改善案検討
 - 3) 手順に沿った手続きの時間測定と発生頻度推定，
作業品質（網羅性，均質性，再現性）の重要性確定とそれらの序列化
 - 4) 3) を基に作業のシステム化優先度（序列化）とどこまでシステム化するかの判断

上記を一般的な能力に置き換えると次のような能力ということになる。

- a) 研究者からの情報取得（項目列挙と定義，データ利用実態把握）
- b) a) の情報の曖昧さ，矛盾の指摘及び真偽を批判的に裏取り
- c) b) 後の情報を構造的にまとめて，汎化
- d) 研究者間，グループリーダとの合意形成

上記 a ～ d の能力は，具体的にどのような能力か以下で具体的に説明をしていく。

- a) 研究者からの情報取得（項目列挙と定義，データ利用実態把握）

⇒研究者と話ができて，実験データの項目及びその定義を聞き出し，また，そのデータをどのように利用しているかの実態を聞き出すことができる能力が必要である。もちろん，「実験データの項目及びその定義を教えてください」，「そのデータをどのように利用しているか教えてください」といったところで，気前よく話してくれる研究者は少ない。本気で興味を持ち，相手が教えてあげようという気持ちになるように仕向ける必要がある。「こういう時困りますよね」とか，「こういう時は，こうすればうまくいったケースも知っていますが・・・」など相手が知らなかった情報を提示できると信頼関係が生まれ，一気に情報量が増えることが多い。過去の様々な経験を常に最前線に提示し，相手の出方で hit 率の高い情報を提示できる訓練を日常的に行っておくとこれら能力は磨かれる。また，「そんなことないけど」と言われる失敗を恐れてはいけない。「そんなことない」ということが分かるということは，非常に重要な情報を得たということであり，自分が一段賢くなったと思ふべきであり，全く失敗ではない。常に勉強させてもらうつもりで，情報取得にあたる心意気が必要である。

- b) 情報の曖昧さ，矛盾の指摘及び真偽を批判的に裏取り

⇒研究者は，悪意を持って嘘をつくことはまずないが，無意識的に嘘になってしまっていることも多いため，「情報が正しいかどうか」は必ず，裏取り（直接的に相手に確認するのではなく，他の情報等から論理的推定で確認する）する必要がある。取得した多くの情報を論理的に検証すれば，矛盾する情報も多々あるはずである。矛盾を解消する条件などを先に検討しておけば，真偽の裏取りにも使える。根本的には，ファシリテータは他人事というスタンスを捨て，明日から自分が代わりに研究を行う（単に決められたデータを探したり，決められたデータ処理をするのではない。実際に実験はできなくとも，データを探して，データを処理，分析をして，他の研究者と議論ができるレベルでよい）つもりで，情報の曖昧さ，矛盾の指摘，真偽の確認を

行えばよい。これらの情報の真偽を直接聞くことなく真偽を確かめる力や矛盾を指摘する力は、探偵や検事に必要な能力でもあり、探偵や検事になったつもりで取り組むといいかもしれない。

c) 情報を構造的にまとめて、汎化

⇒情報を構造的に分類し、まとめ上げる力、具体的な情報から仮説を立てて情報を汎化し、それを検証、改善していく力が必要である。また、それらを自然言語で正確に記述しきる文章化能力も必要である。

d) 研究者間、グループリードとの合意形成

⇒精査された情報を構造的、論理的に説明し、研究者間、グループリードとの合意を取り付けるコミュニケーションおよび交渉能力が必要である。実際に合意を取り付けるためには、論理的に矛盾なくまとめ上げておく必要はあるが、論理一点張りでは難しく、相手が何に重きを置いているのか？などを察知し、それらを天秤にかける形で、交渉に持ち込む力が必要である。

11. データ共有システム運用後陥りがちな落とし穴とそれを防ぐ方策

前項では、「R&D 部門におけるデータ共有システム導入時の落とし穴とそれを防ぐ方策」に関して、説明した。本項では、R&D 部門におけるデータ共有システム運用後陥りがちな落とし穴とそれを防ぐ方策に関して説明する。

1) 利用者がデータを登録してくれない

⇒データ登録はそれ自体は利用者にとってメリットは全くなく、今までより手間がかかるだけの作業であり、忍耐が必要な作業である。過去データの登録などの最初のデータ登録作業は、管理者等の特定の忍耐強い数名が代表して行うことが好ましい。また、データは闇雲に登録をしてもそのあとの活用（データ探査、処理）に効果的ではない。そもそもデータベース化しようという当初目的の具体的な動機部分に立ち返って、また、その動機が概念的であれば、これを機に具体化し、当初具体的目的を達成するために必要なデータを登録対象とすべきである。データの蓄積が進みその効用が明確になってきたときに、その効用を具体例で説明し、徐々に一般利用者にもデータ登録を行ってもらえるようにする。ただ、効用を得るためには、研究スタイルを変える必要も出てくるので、研究スタイルを頑なに変えない人たちには、効用で訴えるより、組織ルールとして、実施するように指示する形が必要である。実際、データベースがなくて本当に困っている研究者はごく一部で、ほとんどの研究者は便利だったら使おうという感覚である。業務系のシステムを見てもわかる通り、システムは便利にするために運用しているわけではなく、属人性からくるブレや揺らぎを排除することが主目的である。省力化の話にしても業務系は、それで人員削減を実際に行うが、R&D はそういう評価の仕方はしないはずである。実際、システムが寄与する省力化は研究の数%にしかならないはずで、それよりもシステムを導入することで質的な変化（AI や機械学習による従来分析で発見できなかった材料開発）を求めているはずである。しかし、システムは手動でできないことはできないという宿命がある。これを乗り越えるには、従来手動でできているデータ探査、データ分析をシステムを利用してできるように慣れるようにし、次に、そのままのシステムを使って、今まで行っていない少し高度なデータ探査、データ分析をする。その次に、少し高度なデータ探査、データ分析の手順化をして、システムの拡張計画を立て、拡張し、……。これを続けていくことで、従来分析で発見できなかった材料開発手順が、AI や機械学習機能としてシステムに組み込まれてくるのである。こういう、step by step の変化は、その step ごとに研究スタイルを変えることになるので、使用方法やデータの解釈は慣れるまでは、全く「便利」ではない。したがって、「便利」を追及している人たちを主対象にして、システム化は進めるべきではない。また、「使用方法やデータの解釈」を十分理解しないまま、システムを使うと間違った他者のデータを使ってしまい、結果として誤った結論を導き出す可能性がある。「使用方法やデータの解釈」がしっかりできない人には、システムは使わせないほうがいいかもしれ

れない。

2) 具体的にどう検索条件を設定すればいいかピンとこない

=>システムを使い始めた頃は、頭の中で「こんなデータを探したい」と思っても、具体的にどう検索条件を設定すればいいかピンとこないものである。手動作業時には、どう探していたかをよく思い出し、それを書き留め、等価なデータ抽出をするには、どう検索条件を設定すべきかを考え、それも書き留めておき、それらを皆で共有することから始めるとよい。どんなことでも文章に落として、それを自分以外の人にわかるように説明することで、自分が何をしようとしていたかが明確になってくるものである。実際には、この作業をしていると、「この方法はよくないかも」、「この方がいいかも」などの考えも浮かんでくる。自分の意識の中だけの考えは、しっかり固まっていると思いがちだが、結構いい加減な部分も多いものである。したがって、システムを使うというのを格好の機会と思って、自分の頭の中を整理し、それを他者と共有しあうことを行っ
てほしい。

3) 検索で指定したい、データ処理をしたい項目が存在しない

=>こういう事態がなぜ発生するかというとシステム開発前のデータ探査方法（データ処理方法も以下同じ）を手順化するときに確認が不十分だからである。どういう不十分さがあるかというと、二系統の不十分さがある。一つ目は、「配合比率で」とか、「評価特性値で」という個別の項目でなく総称的な単語を使った記述をする場合に、その明確な定義を書いていないからである。個別の項目名で手順書を作成すると数えきれない手順書を作る必要があるので、同一パターンごとに汎化した手順書を作るという工程があるが、そこで汎化したときに曖昧性を含む汎化をしてしまっているのが原因である。これを防ぐためには、手順書を汎化するときには、曖昧性を含まないように定義を明確にしておき、そして、具体的な項目名を列挙しておくことが重要である。二つ目は、項目名の合意形成の後にデータ探査手順書は作成することになっているが、手順書作成時に合意形成されていない項目を使ってしまうことがあるからである。項目名の合意形成時には、実験を再現することと評価をしたい値に対する項目を列挙していたので、データ探査のことは考えるようにはしていない。それは、その段階でデータ探査のことまで、考えると意識が発散し、「実験を再現することと評価をしたい値に対する項目」に漏れが発生しやすくなるので、あえてそうしているのである。したがって、手順書を作成するときは合意された項目名リストを見ながら行い、項目名リストにない項目で、データ探査をする必要があるように思ったら、まずは、他の項目でその探査ができないかを吟味し、できるならそれを手順書のTIPS（こういう検索条件を指定したい場合は、こういう条件設定をすれば等価である等）に書き加える。他の項目でその探査ができない場合は、追加項目名の合意形成を行い、項目名リストにそれを追加する必要がある。

4) 検索で指定したい（データ処理したい）項目は存在するが、検索条件として指定（データ処理）できない、しづらい

=>これは、どういう状態の問題かというと、「サンプルIDがIDX0012a～IDX0115までのデータを抽出したい」とか「濃度が1%程度のデータを抽出したい」というデータ探査である。まず、「サンプルIDがID2X0012a～ID2X0115までのデータを抽出したい」は、サンプルIDが数値になっていないので、範囲指定はできない。の目視によりデータを絞り込む場合、人間の認識能力で特定の数字部分だけを無意識に分離をして、抽出している。そして抽出したものが数値でなくとも範囲絞り込みができるように錯覚してしまうのである。手順書を書く時には、「人間の認識能力」を前提にする部分は危険で、単純な機械でもできるレベルに落とし込む必要がある。先の場合は、「特定の数字部分だけ」をどう取り出しているかのアルゴリズムを特定してそれを手順書に書き込むようにするか？、そのアルゴリズム特定が難しければ、特定の数字部分だけ別項目にすればよい。実は、研究でも日常会話でもこういう人間の便利な高度認識能力は、非常に便利でありながら、相

手と意思の疎通がとれないものになっていることも多々あるので、研究のような厳密な扱いが要求されるときは、意識をして、「人間の認識能力」に頼らない表現を心掛けルことが必要である。「濃度が1%程度のデータを抽出したい」は、「1%程度」という部分で、これは本人も曖昧であることは認識していると思うが、手動で行うときはこれでうまくいくということも理解できる。「0.9%～1.1%」とするなら検索条件に指定することは容易である。しかし、そうはしたくないという気持ちもわかる、「0.9%～1.1%」で2点しかなかったり、1万点もあったら困るのである。だから、データを確認しながら分布状況を想定し、範囲を決めているのである。これもそういうところまで手順に書けば、手順書の改善が可能になる。「グラフを作成して傾向把握をしたいために1%付近でデータ点を100点ぐらいに絞り込みたいので、まずはデータ全体の分布状況を確認し、データ点を100点ぐらいになる区間のあたりをつけ、その区間を検索条件に設定する」という感じになる。少し脱線し本課題とは別の議論をするが、「グラフを作成して傾向把握をしたいためにデータ点を100点ぐらいに絞り込みたい」は、データ分析の観点からは、ある意味の意図的データ操作になり、本来はやってはいけないことである。100点に分析的意味はなく、その人の作業労力や見やすいグラフを作成するためのもののはずだ。そういうことを理由にして、分析対象のデータを絞り込むのは行ってはいけないことである。分析結果には、検索条件も添え、なぜその検索条件を使ったかの理由を分析結果と論理づけて説明する必要がある、その時に説明のつかない検索条件は使うべきではないということになる。パラメータの多い実験では、検索条件を調整することで、自分の主張に近い結果を抽出することが可能になるので、たくさんのデータを扱えるようになれば特に気を付けないといけない点である。

5) 検索で抽出されたデータが少なすぎたり、多すぎたりし、適切な数にならない

⇒>ひとつ前の課題でも説明をしたが、検索条件指定し、データを絞り込むと2点しかなかったり、1万点もあったら困るという問題である。一つ目の原因は、すでに話したように「グラフを作成して傾向把握をしたいためにデータ点を100点ぐらいに絞り込みたい」という考えは、データ分析の観点からは、ある意味の意図的データ操作になり、本来はやってはいけないことである。そもそも、適切な数という概念自体がおかしい。今から抽出したい母集団の条件を何にするか？また、なぜ、その条件にするかを問われたときに答えられるように考えをまとめてから検索を行うべきで、できれば、その考えをメモしておき、検索結果の出力に書いておく習慣をつけておくべきである。このような研究の進め方を含め意見交換をする習慣をつけていくと、自分の意図を意識的に認識し、他者に伝える力が身についてきて、自ずとこういう問題は発生しなくなる。

二つ目の原因は、データ数が多くなった時のデータ分析のノウハウや方針を持っていないことである。実際、他研究者のデータまで含んで検索すると思ったより（そもそも他研究者がどういう実験をしているか把握していないわけだからある意味当たり前である）もデータ数が大きくなることは十分あり得る。絞り込み条件が意図したものであれば、むやみにデータ点数を減らすのではなく、そのデータ点数のままで分析を継続すべきである。しかしながら、分析ソフトやグラフソフトで取り扱える上限点数があることも確かである。その場合は、そもそも何を分析したいのかを考えるべきで、そうすることで、一定区間内のデータ点に対して、平均、最大、最小点を算出し、それをプロットすれば済むことかもしれないし、重回帰分析を行ったほうがいいのかも。そもそも、データ点数が増えると多次元の重回帰分析の精度が上がってくるので、データ点数が増えるということは、本来は悪いことではないはずで、それに適した分析ツール、手法を準備することがこの問題の正しい解決方法である。絞り込み条件が意図したもので、データ点数が分析をするのに対して少ない場合は、実験数が少ないということなので、実験を追加すべきということであり、データ点を増やすために検索条件を変えるのは本末転倒である。

6) 項目数が多すぎて、データ分析が実施しにくい

⇒>今までは、実際にデータ分析をする時には、必要なデータがあるファイルや確認が必要な項目部分をファイル内から探しながら、データ分析に必要な項目だけデータ分析をしやすいように別ファイルにデータを纏め

直してははずである。また、データを纏め直している過程で、追加に必要な項目に気づいて、その情報を再度、探したりすることも多いはずである。つまり、データ抽出とデータ分析の前準備が混然一体となっていたのである。それに対し、システムで行う場合は、検索一発で、結果が返ってくるのである。結果を返さない項目はもはや確認はできないし、確認したい項目をたくさん指定すると、膨大な項目が返ってくることもある。しかしながら、いろんなファイルを行き来しながら確認、抽出をすることを考えるとその方が楽なはずであり、ヒューマンエラーも入り込む余地がない。ここで感じる「実施しにくい」は、頭がまだ手動手続きの思考のままなので、システム機能の使用に慣れていないからであり、手動で行っていた手順をシステム手順にしっかりマッピングして、意識をしながらシステム利用をしていくと、慣れていく、つまり研究スタイルが変わっていくものである。実際、慣れる前にシステム利用をやめてしまったり、思い付きのシステム改善を提案して、現行の研究スタイルにシステム合わせるようにしてしまいがちであるが、これでは問題解決にはつながらない。

7) 手動の手順で問題ないので、システムは使わない

=>もし、手動手順で使うデータや項目がシステムの中に登録されていない場合は、まず、それらを登録できるように項目追加などを行うことが最優先である。それと同時になぜ、手順書を作成した段階でそれを逃したのかの原因を調べる必要がある。手動手順調査後に、新たに始められた手順であれば致し方なしだが、過去から高い頻度で使っている手順であれば、調査方法自体の再検討が必要である。そうではなく、システムでも実施可能なのに、あくまで手動手順で行っているのだとすると上述している「慣れ」の問題である。作業品質と作業負担（時間）を客観的に評価し、負荷対品質の高い作業を選ぶ習慣をつける必要がある。システム利用に対する抵抗感が出た場合は、感覚、情緒的に判断を下すのではなく、定量的に判断するように心がけることが重要である。また、システム推進者が一度同じことをシステムと手動で行ってみて、手動より良い点を実際の作業をしながら伝えるなどをすれば改善することがある。実際に、作業品質（網羅性、均質性、再現性）は、システムの方が格段に良くなっているはずであるが、それに関して無頓着な人が多いのも確かである。しかし、グループリーダーや事業部からすると作業品質（網羅性、均質性、再現性）は重要なはずで、この辺りを理解してもらう教育は重要である。実際、手動処理をした結果に関して、システムを使って処理をした結果、結論が変わったり、修正が加わった例を蓄積していくと、作業品質（網羅性、均質性、再現性）の重要性が認知されてくるはずである。R&Dの良くない部分は、徹底的な議論が少なく、グループリーターや事業部に半分あきらめてしまわれている点である。それ故、システム推進者の役割として、このような活動を行うファシリテータの育成が必要となる。R&D部門支援の経験のあるコンサルタントやファシリテータを外注するのも一案である。

8) システムにデータが蓄積され、様々な統計、データ処理機能が提供されているが、結果考察がなく、少し考えればおかしい結論を疑問に思わない人が増えた

=>統計処理、AI処理は、ある程度データ数が大きく、且つそれらの結果と物理的解釈の合意形成ができるまでは、ミスリードリスクが高い。信頼性が高くなってもそれだけから結論を導き出すことは避けるべきである。また、そもそも統計処理、AI処理は高度な結果を示すので、何か意味ありげに思えるが、実際には、根拠となるモデルや説明は提示されない。また、実際に実データはマスクされている分、勝手な解釈が入り込む余地が大きくなる。したがって、統計処理、AI処理が進んでも、実験データを実値のまま検討、分析可能な環境は必ず必要で、実値データの確認は必ず行い、自分およびグループメンバーで共有し、仮説、検証のloopを回すことが重要である。

9) データ活用を専門とする MI (Materia Informatics) 部隊は、データ分析結果はどんどん出してくるが、研究開発推進にはあまり貢献していない状況になっている

⇒ AI がよく使われる画像処理などは、情報を捨象なしで処理をしているが、各ピクセル情報は正しく、入力サンプル数も大きくすることが可能である。もう一つの AI 成功例のマーケティングなどの社会分析は、個々の詳細な情報を捨象することで、情報次元に対しての入力サンプル数を大きくし、集団としての統計的特性を利用することで、AI を成功に導いている。一方、物理、化学を活用した材料、製品開発は、情報次元数に対して入力サンプル数は圧倒的に少ない。また、個々の詳細な手続きは意識的に行っているものなので、情報を捨象した統計処理をすべきではない（もちろん、分析の結果、その情報が不要だったということはあるが・・・結果としてそうなるだけで、最初から捨象してはいけない）。また、実験を再現するだけの十分な次元の情報が記録されているかどうか？記録したとしても正しく記録されているのか？など、情報品質も不十分なことが多く、情報品質の改善をかけながら MI を進めていく必要がある。つまり、MI 部隊は、実験者の実情を知って、実験者に実験の様々な面で改善提案をできるだけの信頼を得られる存在でなければならない。MI 部隊は、システム構築前の実験者ヒアリングにも緊密に関わることが重要で、そのためには単なるデータアナリスト集団ではだめで、同じ分野での研究現場経験者が 7,8 割程度、また、それらを取りまとめるファシリテータも含まれる必要がある。

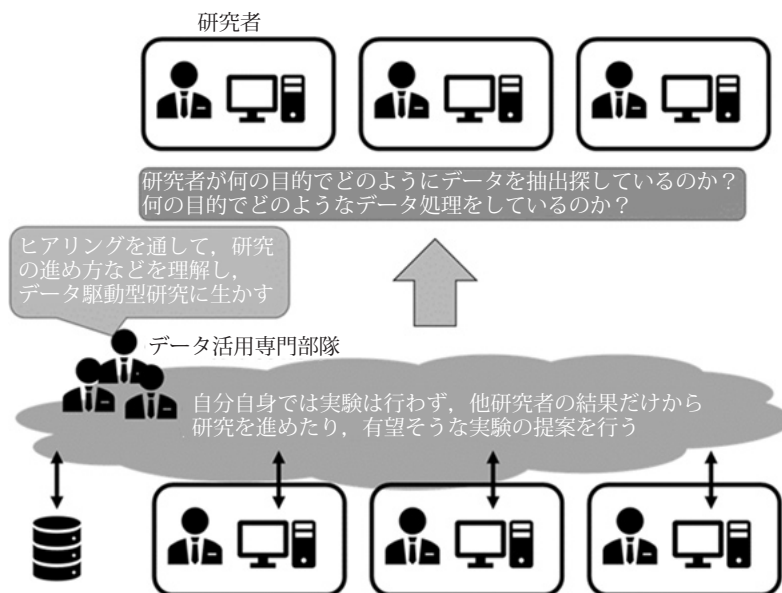


図 16 データ活用専門部隊によるデータ探査、処理実態調査

12. 効果的なデータ共有・利活用手法と運用体制の作り方

前項では、「データ共有システム運用後陥りがちな落とし穴とそれを防ぐ方策」に関して、説明をした。本項では、利用、活用されるデータ共有システムを実現する為の効果的なデータ共有手法と運用体制に関して、具体的な対処方法を提言する。

a) 「実験及び分析を再現するのに十分な項目」の調査、分析

研究者自身が「実験及び分析を再現するのに十分な項目」を意識でき、それをグループなどで話し合い、合意形成を可能にする状態にできるのが一番好ましいのだが、研究者の特性上、難しい場合が多いのも事実である。そのような場合は、研究グループ内に支援者という名目でファシリテータを配置し、「実験及び分析を再現するのに十分な項目」の調査、分析を行なわせる体制を作ることをお勧めする。「実験及び分析を再現するのに十分な項目」がわからない、再現性の難しい「実験及び分析」の中にはあるはずだが、ここでいう「実験及び分析を再現するのに

十分な項目」は、研究者自身が意識的に制御できていないものも含めて追及するという意味ではなく、意識的に制御しているものだけを意味する。もちろん、「実験及び分析」の理解が進むことで、意識的に制御できるものが増えてくるはずで、そういうものは随時追加できるような取り組みをすることが重要である。

b)「現状どのように過去データを探しているか？」の調査、分析

研究者自身が「どのように過去データを探しているか？」を意識でき、それをグループなどで話し合い、合意形成を可能にする状態にできることが一番好ましいが、研究者の特性上、それが難しい場合が多いのも事実である。その場合は、研究グループ内に支援者という名目でファシリテータを配置し、「どのように過去データを探しているか？」の調査、分析を行なわせる体制を作ることをお勧めする。もちろん、研究の進め方が変わることによって、「どのように過去データを探しているか？」も変わってくるはずで、そういう変化に追従できる取り組みをすることが重要である。

c)「蓄積されたデータをどのように利用するか？」の調査、分析

「蓄積されたデータをどのように利用するか？」は、データ共有システムを作る前に、入念に調査する必要があることであるが、システム構築をする関係者が研究者自身に直接ヒアリングをしても多くの場合、適切な情報が入手できない。研究者は、自分が現在どのようにデータを利用しているのか？、データが蓄積されたらどう利用したいのか？を意識的に考えていることは稀で、多くの場合漠然としたイメージでしか意識できていない。また、システム構築の関係者は、システムを構築することに前のめりになるため、研究者の言葉を直接的に鵜呑みにしてしまうことが多く、本当にその利用方法で研究者の課題が解決するのかまでの考察ができない。実際、研究者が行なっていることを、システム構築関係者が理解、把握することが専門領域的にも難しいのも事実である。したがって、前項で述べた研究グループ内に支援者という名目で活躍できるファシリテータ的な人材を調査、分析の任につけることも検討する必要があるのである。さらに、本調査、分析はデータ共有システムを作る前だけでなく、システム運用後も継続し、システムの改善、改良に役立てる必要がある。R&D部門は他の業務部門と異なり、多種多様、経時変化していくものなので、そういう変化に追従できる取り組みをすることが重要である。

d)「実験及び分析を再現するのに十分な項目」を確実に記録させる体制

グループ内で「実験及び分析を再現するのに十分な項目」を記録する重要性について話し合いの場を持ち、自律的に確実に記録できるようになるのが一番良いのだが、そうならないことの方が多いはずである。そのような場合は、研究グループ内に支援者という名目でファシリテータを配置し、「実験及び分析を再現するのに十分な項目」が記録されているかどうかを確認し、足りなければ研究者に追記を依頼したり、ヒアリングをして自らが追記する体制を作ることをお勧めする。MI部隊がこのファシリテータ役を担うことも効果的な案である。実際にこのような体制を採ることは、誤記録や無意識の記録漏れ、データ捏造の抑制や実験及び分析の進捗状況の把握にも有効である。

海外では、徹底した分業体制を採り、分業間でデータ引き渡しをするポイントにおいて責任の所在を明確化することで、各自の責任部分の項目を記録する確実性を担保している。この方法は、分業部分毎の作業を明確化することで、業務の標準化と人材の配置転換の容易性を高くできる半面、各人の創意工夫の自由度や経験値を十分に生かすきれないというデメリットもある。日本の多くの企業では、良くも悪くも責任の所在の明確化や徹底した分業をせず、各人の創意工夫の自由度や経験値を十分に生かした業務の取り組み方法で、効果的なR&Dを進めてきた。しかし、「各人の創意工夫の自由度や経験値を十分に生かす」ために、データ共有が属人的になってしまっており、昨今のデータが大量に発生するR&Dでは属人化のデメリットがメリットを上回ってきている。

ファシリテータを配置しても改善できない、ファシリテータ自体を配置できない場合は、海外のような徹底した分業体制を取り、責任の所在の明確化と責任追及をする体制への移行も考える必要がある。

e) システム利用を推進する体制

単に「利用してください」とお願いをしたり、「どんな機能が必要ですか？」とヒアリングをしても、効果的にシステム（蓄積されたデータ）の利用が推進されることはない。上述したが、研究者は研究対象について深く考えるが、自分が現在どのようにデータを利用しているのか？、データが蓄積されたらどう利用したいのか？を意識的に考えていることは稀で、多くの場合、漠然としたイメージでしか意識できていない。そのことを前提にして、システム利用を推進しようとするのであれば、研究者のどのような課題がシステムで解決するかを考え、課題がどのように解決できるかを提示し、利用へ誘導していく必要がある。また、その過程で、現在のシステムで解決できない課題が分かり、それを解決するためにはどのような機能があるべきかを考えることで、システムの拡張方針が決まることもある。

システムの利用推進や運用に関する体制には、もう一つ注意が必要な側面がある。運用者が研究者の行なっていることを理解、把握できなければならないことは言わずもがなだが、理解、把握をしていたとしても、業務が完全に分離されている状況では、次第にそれぞれの立場に偏ってしまう。その結果、研究者は重要度が低い機能を要求し、運用者はそれを漫然と受け容れるか、完全に否定的な態度をとってしまう状態に陥りがちになる。これらの状況を打開するためには、人事交流による役割のローテーションが有効である。研究者を推進運用側に異動したり、兼務させることで、相手の立場、観点を意識できるようになり、特定の役割に偏り過ぎた認識が抑制され、効果的なシステム改善が可能になる。

また、システムの利用を推進するためには、運用体制だけでなく、利用者の体制に関しても工夫が必要である。データ共有ができていない状況では、当然ながら研究者は自分以外の人が生じたデータを使うことを前提として、研究を進めていない。しかし、データ共有できる状態になり、いくら便利になったとしても、急に研究の進め方を変えられるわけではない。もちろん、一部の好奇心が旺盛な研究者は、これを機に研究の進め方を変えるかもしれない。研究者は研究対象に対して好奇心は旺盛だが、それ以外の部分に関しては、人並み以上に保守的な人が多いのも事実である。したがって、せっかく構築したデータ共有システムを最大限効果的に活用するためには、従来の研究から離れて、他の研究者のデータのみで、研究を行なう専門部隊を立ち上げ、システムを最大限効果的に活用し、成果を出すことを彼らの使命にするのである。彼らは、自身で実験を行なわないので、自分たちの成果をあげるために、有望そうな実験の提案やそのための説得資料作り、予算取りなども自律的に行なうようになるはずである。

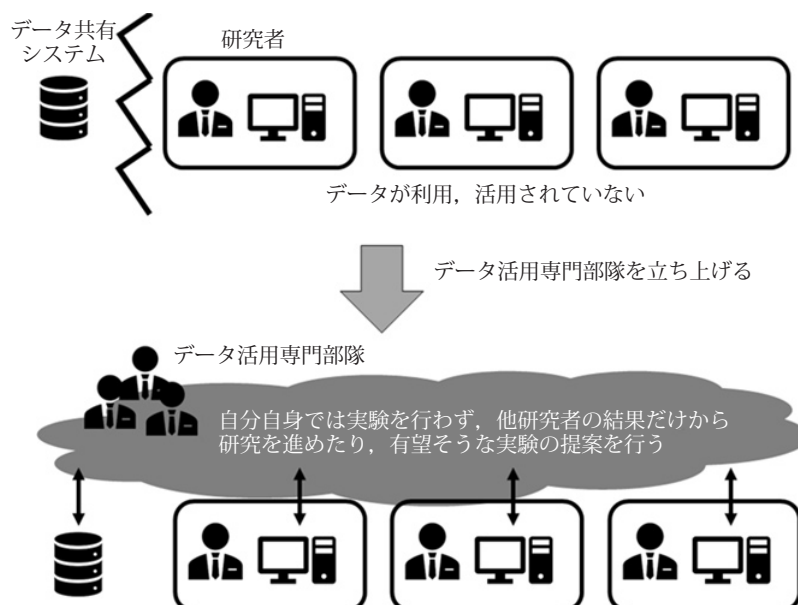


図 17 データ活用専門部隊によるデータ活用と既存研究者の活性化

おわりに

本節では、R&D 部門におけるデータ共有の実情が属人的、形骸化したデータ共有となっている実態を紹介し、その問題点及び原因を検討した。それらを基に効果的なデータ共有手法と運用体制の作り方に関して提言をした。読んでいただいてお分かりだと思うが、R&D のデータ共有を行なうということは、何かを買って、解決するという問題ではなく、企業風土及び R&D 従事者の意識変革が必要である。R&D 従事者は、業務内容が極めて専門的で、社内でも対話する機会が少ない人たちだと思う。研究者は、とっつきにくいところはあるが、地頭は良く、非常にまじめなので、これを機に話をしてみれば、仲良くなれるかもしれない。そういう人とのつながりを作ることが、R&D 部門におけるデータ共有の第一歩である。

文 献

- 1) 川田重夫, 田子精男, 梅谷征雄, 南多善, 上島豊他, PSE book— シミュレーション科学における問題解決のための環境 (応用編), 川田重夫, 田子精男, 梅谷征雄, 南多善 共編, 培風館, (2005), p69-82
- 2) 谷啓二, 奥田洋司, 福井義成, 上島豊, ペタフロップスコンピューティング, 矢川元基 監修, 培風館, (2007), p183-202
- 3) 牧野圭一, 上島豊, 視覚とマンガ表現, 臨川書店, (2007), p1-5,221-229
- 4) 上島豊, 月刊「研究開発リーダー」8月号, 技術情報協会, (2020), p33-37
- 5) 上島豊, 月刊「研究開発リーダー」9月号, 技術情報協会, (2020), p53-57
- 6) 上島豊, 月刊「研究開発リーダー」1月号, 技術情報協会, (2022), p58-63
- 7) 上島豊, 月刊「研究開発リーダー」2月号, 技術情報協会, (2022), p46-50
- 8) 上島豊, 月刊「研究開発リーダー」3月号, 技術情報協会, (2022), p62-65