SEMINAR REPORT

ＩＴ運用業務に革新をもたらすための自動化とは？

株式会社日立製作所
サービスプラットフォーム事業本部
IoT・クラウドサービス事業部
事業推進本部
主任技師
加藤 恵理 氏

ご紹介いただきました日立製作所の加藤です。私の方は、「IT運用業務に革新をもたらすための自動化とは？」というタイトルでお話させていただきたいと思います。

さまざまなシステムがあって、それには開発等を伴いますが、その先に必ず運用というフェーズがあります。運用は、そのシステムが稼働している限り継続しなければいけない宿命を背負っていますが、その運用業務をいかに軽減するか、何かお手伝いすることはないかという観点からお話させていただきたいと思います。

ＩＴ部門に求められる役割の変化

1．ＩＴ部門に求められる役割の変化

最初に、IT 部門に求められる役割の変化について、今 IT 部門が直面している問題には何があるのかという側面から、課題を掘り下げてみたいと思います。

世の中の動向を少し別の観点から考えてみると、全てのビジネスがデジタル化すると言われています。これは、皆様方もよくご存知のことと思いますが、最新のテクノロジーにより、実世界がさまざまなデジタルデータで捉えられるようになってきています。その結果、社会や事業の変革が急激に進んでいるように思います。

IoT というキーワードはよく耳にしますが、これは単に機器がネットワークにつながってデータを出すということに留まらず、さまざまなシステムが連携、再編されて新しい価値を生み、世の中の仕組みが変わっていくことにつながるものと考えています。そのような世の流れの中で、デジタル技術を用いてイノベーションを創出する新興企業の台頭があって、既存企業も今までの業界の垣根を超えて、これら新興企業との戦いに直面せざるを得ない状況にあるように思います。

ビジネス自体がデジタル化することで、顧客に提供するサービスが高度化します。もちろん、情報システム部門にとって、社内ユーザも顧客の内に入ります。

色々なところでサービスというキーワードが出てきますが、高度化するサービスは、使いたいときに使える、一人一人のニーズに合わせたサービスでないといけません。

スマートフォンユーザは増大しています。この増大の勢いからすると、サービスの利用者は右肩上がりに増えて行くことになります。このような顧客ニーズに素早く対応して、サービスやサービスを支える業務システムを変えていく必要があります。今までのようにゆっくりと、上流工程でしっかりと設計してから動かすようなことをしていたのでは、とても間に合わないことになってしまいます。

一方で、ビジネスにおける IT への期待はどうなのかというと、ここ数年、特に日常生活を考えてみるとお分かりになると思いますが、消費者が意識しないうちに、さまざまな IT を使ってサービスを受けているケースが急激に増えているような気がします。

図１には、金融決済、緊急災害情報、宅配など、ごく当たり前のサービスを挙げていますが、これはほんの一例に過ぎません。要するに、消費者の行動、顧客の行動がビジネスに大きなインパクトを与えるようになってきたということで、かなり前から言われていることですが、IT はビジネスに欠かせない存在であって、IT に対する期待は一段と大きくなってきています。

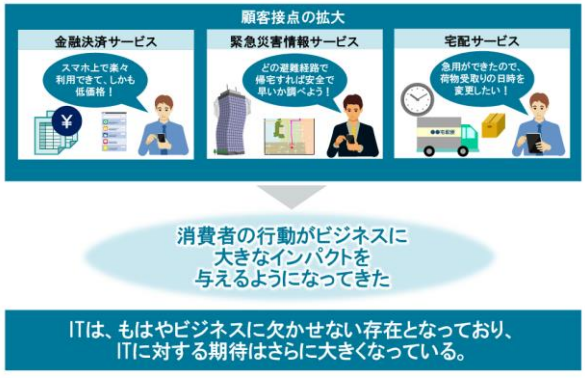

図１　ビジネスにおける IT への期待

そのような状況で、企業がビジネスのチャンスを逃さないためには、ビジネス環境の変化に迅速、柔軟に対応できるITシステムと、その運用が不可欠になるということです。

IT運用で重要となる特性要素は何かというと、ITに任せられるところはITに任せるということで自動化して、システム全体ではなくビジネスへの影響に対して見える化することにあると思います。さらに、変化に応じて迅速に柔軟に対応するための俊敏性と、無駄もリスクも極限まで減らした最適投資が必要となります。そして、ビジネスの規模に応じた伸縮自在なITということで、調子がよいときはさらに拡張するが、悪くなったときは速やかに止められることが強く求められています（図2）。

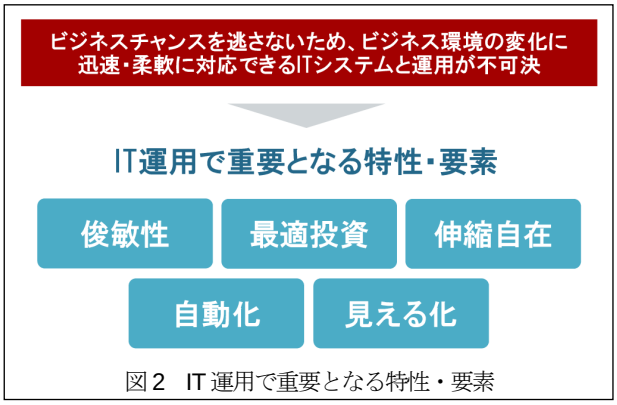

図2　IT運用で重要となる特性・要素

このようなIT運用に必要な要件をひも解いてみると、一つには、変化に強い柔軟な対応ができることが挙げられます。次は、さらなる安定稼働です。安全稼働により故障は少なくなりますが、集約されたシステムでは、何かが起きたときに、影響範囲が非常に大きくなってしまうので、今まで以上に安定稼働が求められます。そして、起きてはほしくないのですが、万が一問題が発生したときには、複雑化したシステムに対してどれだけ迅速に対応できるか、ということが挙げられると考えています（図3）。

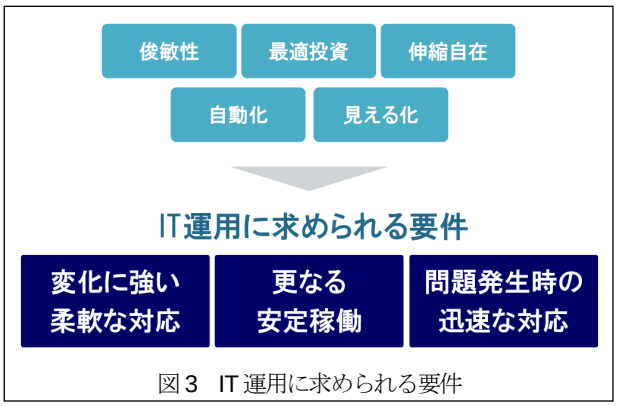

図3　IT運用に求められる要件

ＩＴ運用の現場における課題

ここまで、ITを取り巻く環境についてお話しましたが、それでは、実際のIT運用の現場における課題は何かということで、皆様方と認識を共有したいと思います。要するに、ITをうまく使いこなせればよいのですが、実際のところ一筋縄ではいかないです。さまざまな課題をいかにして解決するか、このことを考えていきたいと思います。

柔軟な対応をしていく上での課題となると、仮想化やクラウドの普及に伴って、クラウドの特性を生かして、運用コストを最適化できる柔軟な業務運用が求められることになります。

近年、クラウドが大いに活用されていますが、クラウドに何が求められているかというと、従来のオンプレミスのように設備を全て自前で調達するのではなく、使いたい分だけのリソースを借りて始められることにあると言われています。しかしながら、業務量に応じた柔軟なリソース運用が結構難しくて、実のところ、ビジネスの変化に対応できていないという課題に直面しています。

この課題は、仮想化技術が出てきたときにも言われたことなのです。仮想化技術を使うとすこぶる便利で、おそらくバラ色の状況を思い浮かべられたことと思いますが、実際、仮想化そのものは便利なのですが、却って運用コストが上がるという事態を招いてしまったことにあります。

同じように、クラウドが出現して、クラウドが出てきたゆえの課題がたくさん出てきて、これを解決する必要が生じてしまいました（図4）。

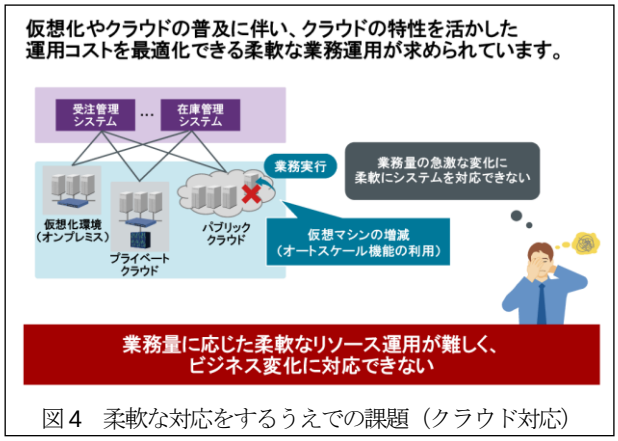

図4　柔軟な対応をするうえでの課題（クラウド対応）

最近は、ウェブベースのアプリケーションの採用が進んでいることもあって、他のシステムから既存の業務システムへの接続も、ウェブインターフェースが主流になってきています。しかし、従来はガチガチの専用インターフェースを使って、堅牢に業務システムを組むのが当たり前のことであって、企業はその類の財産をたくさん保有しています。他のシステムをどのようにしてつなぐのかという課題があるのです。つまり、独自のインターフェースではシステム連携が困難で、業務の拡張が煩雑になってしまって、ビジネスチャンスを逃してしまうことになるのです（図5）。

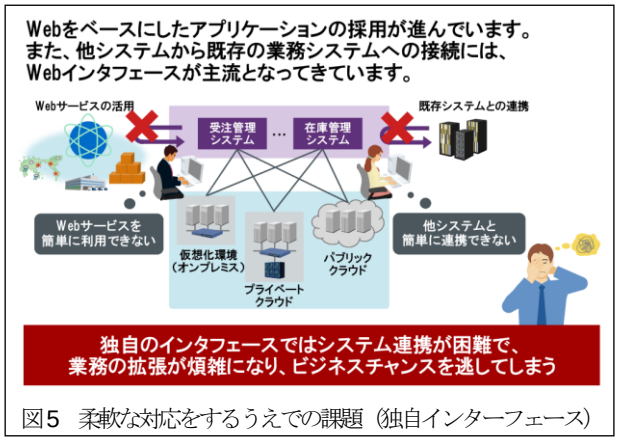

図5　柔軟な対応をするうえでの課題（独自インターフェース）

2 番目は、安定稼働する上での課題です。システムを統合したことで、監視対象は増大して、さらに複雑になってきています。そもそも、監視業務を始めるための準備が大変なのです。監視業務というのは、業務を安定的に動かすために付随的に行うものであって、大事な業務ではありますが、新たなビジネスを育てるような業務ではありません。

システムが統合されると、監視対象が増大して、当然ながら、イベント通知がたくさん上がってきます。大量の通知があると、重大な通知を見逃してしまう恐れがあります。さらに、複数のシステムから色々な形式のメッセージが通知されるので、これを逐一理解するだけでも大いに時間がかかってしまい、複雑化、多様化した IT システムの安定稼働を継続するのが非常に難しくなってきています（図 6）。

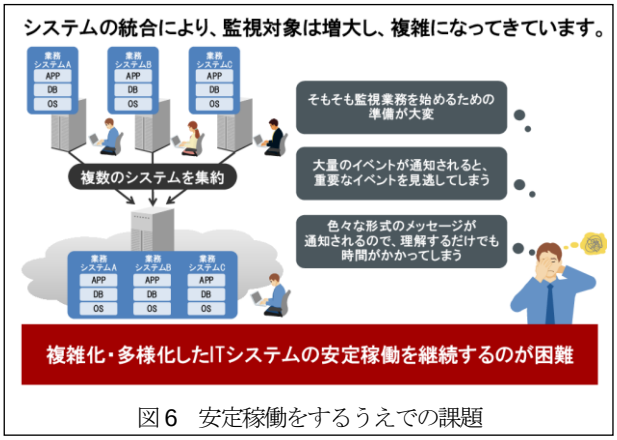

図 6　安定稼働をするうえでの課題

さらに、3 番目の課題です。最近市場では、抜本的なコスト削減に向けて、仮想化やクラウドを利用した IT 基盤の統合、再構築を加速しています。とにかく再構築すれば、コスト削減になるだろうということです。

ところが、集約によりシステムが複雑になってしまいます。ですから、障害発生時に何から調査すればよいのかわからないこともあって、システムが集約されたことによる障害発生時の影響範囲の拡大、復旧作業の長期化が生じて、これがビジネスへの甚大な影響を与えているのは間違いない事実であると思っています（図 7）。

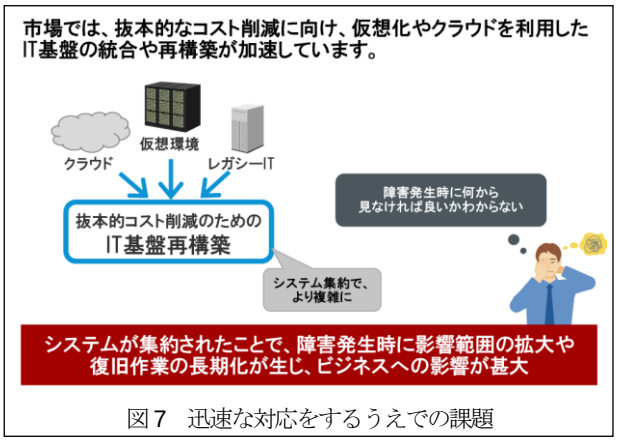

図 7　迅速な対応をするうえでの課題

図 8 は、IT 運用の現場における課題 3 つのまとめです。左側が「IT 運用に求められる要件」で、中央がその要件を実現するための「運用変革に向けてのアプローチ」です。実現に向けて色々と手を打つと、運用現場でさらなる課題が出てきてしまうといった、何かイタチごっこのような状態となっています。これを解決することが急務となっています。

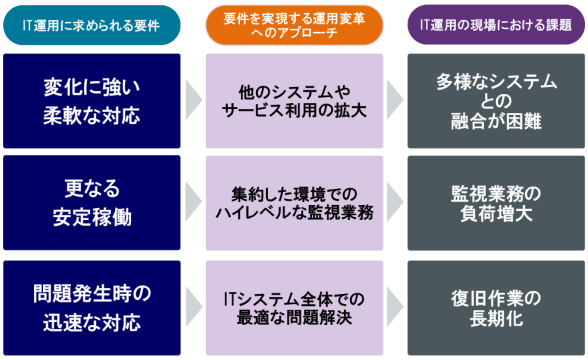

図 8　IT 運用の現場における課題（まとめ）

ＩＴ運用をスマートに実現するポイント

やらなければいけないゴールが明確になってきて、それを阻害するものも少しずつ見えてきました。まずは、実際に IT 運用を実現していく上でのポイント、現状できている範囲をお話したいと思います。

(1) 柔軟な対応を実現するための要素

最初に、柔軟な対応を実現するための要素について考えてみます。先ほど、IT 運用の現場における課題の中に、多様なシステムとの融合が非常に難しいものがあると述べました。

色々なシステムが連携するときに、それぞれ OS 独自のインターフェースを使用していたのでは容易に接続できないということで、OS に依存しない HTTP 経由の汎用インターフェース（REST API : REpresentational State Transfer API）を用いることで、業務運用の幅を広げられるようになります（図 9）。

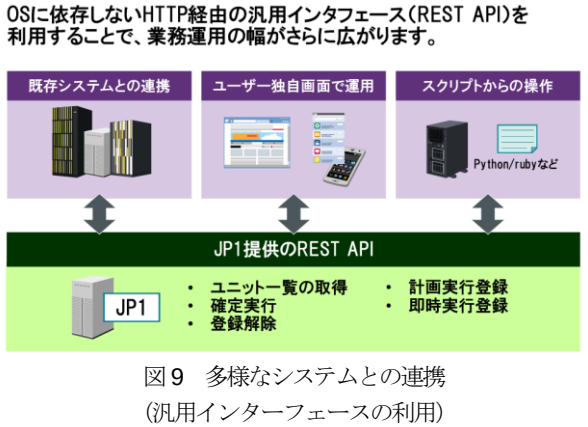

図 9　多様なシステムとの連携
（汎用インターフェースの利用）

企業はさまざまな資産を持っています。それらを全て捨てて、新しく構築するということなら、対応は至って簡単です。しかし、現実は既存のシステムをうまく生かしながら、次にステップアップしていくことになります。それゆえ、既存システムと連携する必要があります。また、ユーザ独自の画面で運用していることもあります。さらには、スクリプトコマンドを使ったシステム操作は、未だに運用管理者の間では用いられています。

彼らはGUIなど使いません。

今はこのように状況ですが、システム運用を容易くするためには、REST APIを使った運用管理業務をいかに早く、いかにスムーズに導入するかに拠るところが大きいのではないかと考えています。

多種多様なシステムとの連携の必要性を挙げましたが、単にシステム同士をつなげるのではなく、色々なシステム、サービスから提供されるデータをうまく取り込むことで、多様なビジネス価値の創出が可能になってくると思います（図10）。

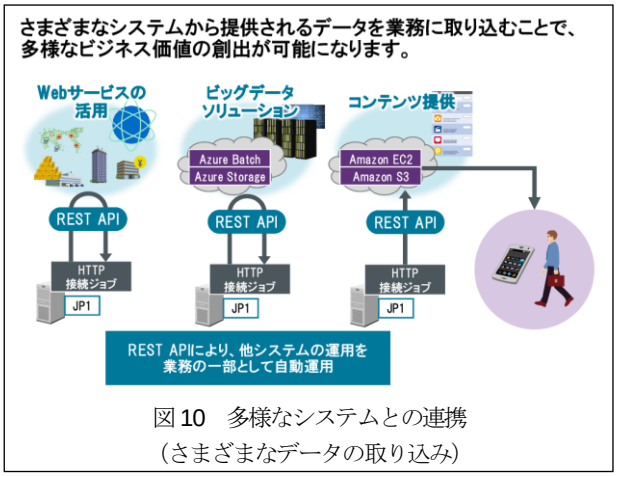

図10　多様なシステムとの連携
（さまざまなデータの取り込み）

今は無料の情報提供サービスがたくさんあります。ウェブサービスの活用や、ビッグデータソリューションという形でのデータの利活用など、皆様方も色々と考えておられると思いますが、これらをうまく取り込むことです。そして、スマートフォンやタブレットに代表されるように、さまざまなコンテンツをうまくつなげることで、素早く業務システムをつくって自動運用することが、今では実現可能となってきています。

ここで登場してくるのがREST APIです。これを使うことで、他のシステムで運用されているサービスを、あたかも自らの業務の一部として自動運用することが可能になります。

従来、外部サービスのところも全て自らつくるのが当たり前のことでしたが、今はことのほかよいサービスが提供されています。このようなサービスと組まないことには、新興企業のスピードに全くついて行けません。こういった点をしっかりと考えていく必要があると思っています。

図11は、多様なシステムとの連携の例です。コンビニエンスストアを展開している企業では、気温、湿度、風などの天候による過去の商品販売数の統計データをうまく利用して、最適な配送を実現しています。

雨が降り始める前に、コンビニエンスストア、駅の売店、他の店でも、傘が山のように積まれているのを目にしたことがあると思います。以前に比べて今の天気予報は、何時間も先ではなく、ほんの数分先でも読めるようになって、すぐに傘を準備できるようになったからです。

図11の例では、翌日の天気を取得となっていますが、リアルタイムのデータを取ってきて、次に何が起きて、その後人はどのような行動を取るのか先読みして、ビジネス展開を図るというようなこともできると思います。

無料で精度の高い天気情報が次々と出てきています。そのような天気情報と、今までの統計情報と、このようなときには何が起きているかという情報を基にして、今すぐや翌日に並べる商品の種類や数を調整するという、ビジネス直結の業務が各店舗では大いに必要となって、このようなシステム連携が活用されるのです。

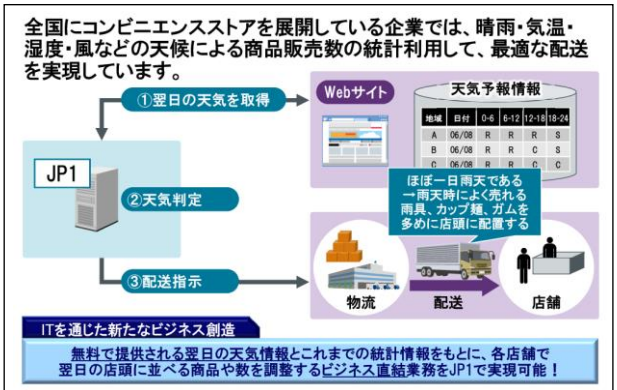

図11　多様なシステムとの連携（実施例）

雨が降れば傘が売れるというのは、当たり前のことです。1日中雨の場合、もちろん雨具は売れますが、カップ麺やガムなども売れるという人の動きがわかっていれば、傘だけではなく、それに付随して購入してもらえそうなものを一通り揃えて配置することが、現状ではごく当たり前に行われています。

次は、クラウドサービスとの連携についてです。クラウドの特性は、使いたいときにすぐ使えて、使い終わったら直ちに切り離すことで、業務の実行環境を都合よく構築して、業務量の急激な変化にも柔軟に対応できて、それで、最終的にはコストの最適化が図れるというものです（図12）。

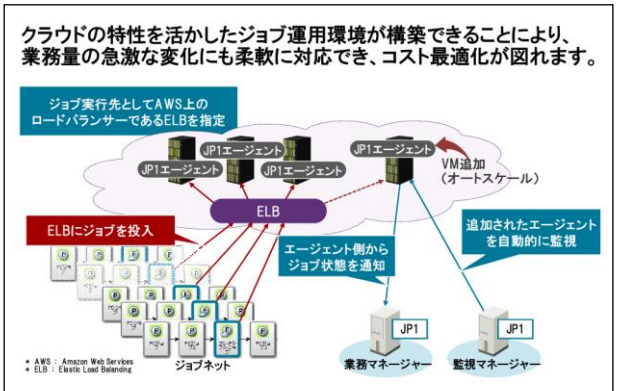

図12　クラウドサービスとの連携

従来は、エージェントをきちんと配置して、マネージャーが監視下にあることを認識しないと監視業務は始められなかったのですが、エージェントが追加されたことを自分のマネージャーに通知をすることで、監視業務がスムーズに始められるようになります。要するに、その業務を動かすために追加されたものが監視対象から外れてしまうと、そこで何か問題が起きたとき、問題が起きたことに気づかないということが起こり得るので、うまく連携することで、監視されるようにしています。

(2) 安全稼働を実現するJP1

次に解決すべき課題となると、IT運用現場における課題の中で、安定稼働の実現が必要となりますが、その監視業務の負荷

が増大してきています。

安定稼働を実現するためには、当然、監視が必要となります。しかし、監視業務の負荷が増大してくると、運用現場の方たちからは、これ以上監視対象を増やさないでほしいという要望が上がってきます。

そもそも監視を始めるには、準備にものすごく時間がかかります。監視のデフォルト設定は、システムの重要度によって変わってきますが、運用現場ではマニアックというか、とてもきめ細かく閾値の設定をする傾向にあります。

弊社の JP1 という統合システム運用管理ソフトウェアは、20 年ほどの歴史がありますが、顧客の要望を全て取り入れて設定しようとすると、ものすごい数の監視設定情報を入力しないといけなくなってしまいます。ところが、実際に現場で聞いてみると、100 項目、200 項目とある設定の多くはデフォルト値のままで使っていて、3 分の 1 ほどしか設定を変えていないことがわかりました。

そこで、最近の取り組みとして、顧客にきめ細かく設定してもらう方法もありますが、幾つかはデフォルト値が設定してあるエクセルのパラメータシートを用意して、まとめて設定できるようにしました。そうすると、構築にかかわる時間を削減できるだけでなく、人が設定することによる転記ミス、入力ミスの軽減にもつながります。

システム定義を行うときには、さまざまな設定パラメータの定義書を起こします。従来、監視対象のサーバに対して一つ一つ設定していたと思いますが、この定義書をベースにスクリプトを実行して、自動的に設定できるようになります。これで、漏れなく正確に自動反映できます。設定は、サーバが 1 台やそこらなら緩々とできますが、何百台全てに同じ設定をしなければいけないとなると、それだけで気が萎えてしまいます。

容易く設定できるようにするためには、自動化が一つのキーワードになるのではないかと思っています（図 13）。

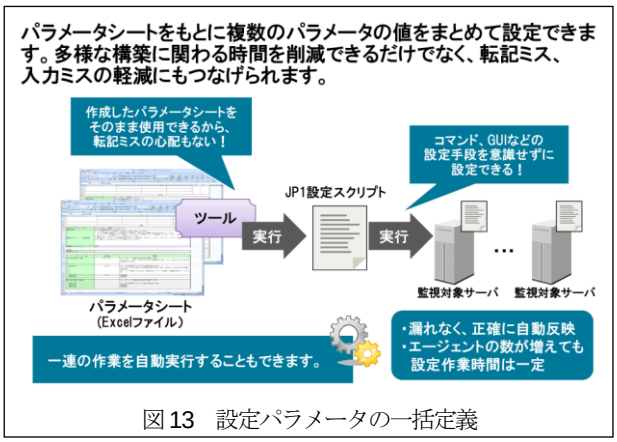

図 13　設定パラメータの一括定義

色々なシステムを集約して監視すると、システムログだけに限らず、さまざまなメッセージ、ログがものすごくたくさん上がってきます。そうなると、フィルターにかけて目的のイベント情報だけを絞り込んで画面に表示するようにしておかないと、あっという間にログ表示がスクロールされてしまい、現場のオペレーターが見逃してしまうようなことが起こり得るのです。

このようなことが現場では起きているということで、オペレーターが見なければならない特定の情報に、色づけしたり強調表示したりして、特定のアプリケーションやホストからのイベント情報だけを表示するように絞り込みします。もちろん、最終的には記録として色々と残さなければいけないので、特定のログしか見えていないというのはよろしくない話なのです。たくさんあるログをいかにしてオペレーターが見逃さないようにするかは、安定稼働につなげるための課題の一つではないかと思っています（図 14）。

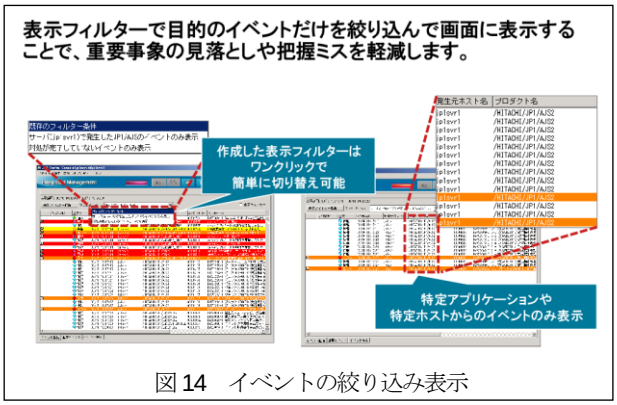

図 14　イベントの絞り込み表示

さまざまなシステムから色々なメッセージが上がってくると、オペレーターは何が書いてあって、どこを見ればよいのか、それこそメッセージによって千差万別なので分かりづらいです。手順に従って行動し、マニュアルどおり対応するオペレーターにとって、難解で不要な情報がたくさん含まれているメッセージは、視認性がよくないです。そこで、さまざまなシステムから上がってきたメッセージを変換して、あたかも 1 つのシステムでつくられたかのように見せる必要があります。これで、問題の内容と 1 次切り分けがし易いようなフォーマットのメッセージに変換して表示されるので、オペレーターの視認性が向上します（図 15）。

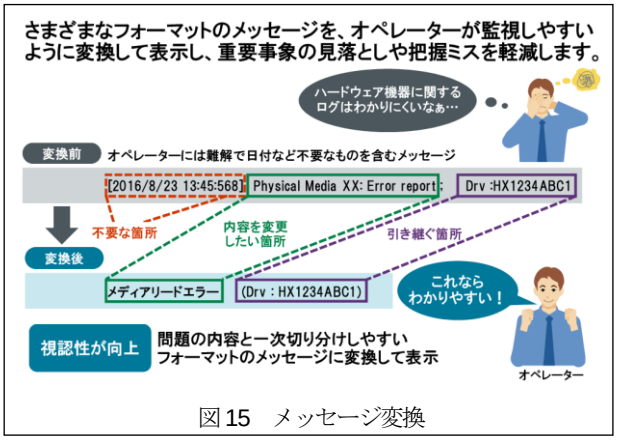

図 15　メッセージ変換

続いて、大量イベント発生時の自動検知、自動集約、集約表示についての話です。

繰り返されるイベントの監視を抑止するために、表示するメッセージ数やイベントに連動するアクション数を削減することで、重要事象の見落としや管理者の負担が軽減されます（図 16）。

エラーで大量のイベントがいきなり上がってきたときは、同じエラーメッセージばかりというイベントストーム状態になっています。エラーメッセージに対してアクションが 1：1 となっていた場合、エラーのたびにメールが通知されて、そのエラーメッセージが収まるまで、何百通、何千通ものメールを受け

取らなければいけないことになってしまいます。

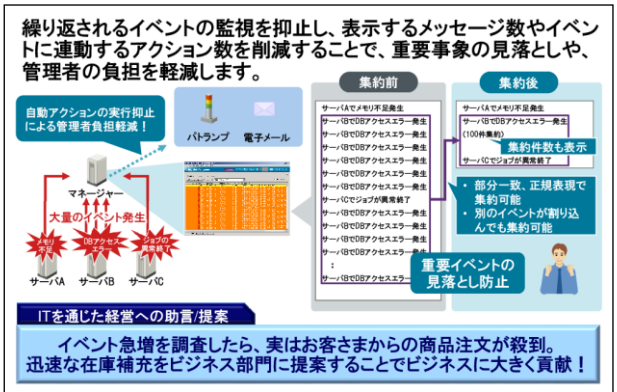

図 16　大量イベント発生時の自動検知、集約表示

イベントを集約すると同時に、そのイベントに対して何らかアクションを起こすことも集約させて、イベントそのもの、すなわちメッセージそのものを集約するところまでは、先ほどの視認性の向上につながります。さらに、メッセージを受けた管理者がアクションを起こすところも絞り込みが必要になってきます。ただし、イベントが急激に増えたことが必ずしも障害ではないという認識が必要です。例えば、イベントが急増して調査してみたら、顧客からの注文が殺到していたようなケースがあり得ます。

イベントが増えたときは、通常、何か問題が起きていると考えることが多いのですが、正常なイベントが増えている、何らかのビジネスチャンスが生まれている可能性があります。これらをきちんと見分けることで、例えば、注文殺到のケースでは、迅速に在庫の補充を行うことで、ビジネスチャンスを取り逃がさなかったというケースもあるのです。イベント発生時は、エラーメッセージを集約するパターンが多いのですが、何ゆえそのイベントが増えているのか、前向きにきちんと分析することで、ビジネスチャンスの到来ということもあり得ます。

IT 部門の監視業務は、システムを安定稼働させて、悪いところを早く見つけて、障害が起きないように監視することが主要な役割でしたが、最も間近でイベントがわかる立場にいるのだから、監視だけでなく、自らもビジネスに貢献する形で参画できるのではないかと思っています。

(3) 迅速な対応を実現する JP1

迅速な対応の実現に障害となるものは何か。IT 基盤が集約されたことで、却ってシステムが複雑になってしまい、何かが起きてしまうと、その復旧作業が長期化してしまうことだと思います。

システム運用において、障害のリスクそのものは軽減できると思います。障害を完全に無くすことは甚だ難しいと思いますが、ゼロに近づけることはできると思います。それゆえ、万が一障害が発生しても、迅速な対応がビジネスに対するインパクトの軽減につながると考えています。

図 17 は、障害復旧のサイクルです。障害が起きたら、現在のシステム構成を手動ではなく、自動で把握します。次は、障害の発見と緊急度の判断です。この判断を誤ると、本来やらなければならないところが後回しになって、問題が拡大、深刻化してしまいます。これはよくあることです。次は、業務の影響への確認です。間近で起きていることは、誰でも対処すると思いますが、このまま放置しておくと他の業務に影響が出るものもあります。この確認が必要になってきます。最後に、障害分析と復旧対処です。分析と対処、どちらから先にするか。障害パターンがわかっていて、発生したときの対処方法が見えている場合は、分析は後回しにして、まずは復旧から始めるという方法もあるでしょう。

障害復旧のサイクルは、このようにモデル化できると思います。

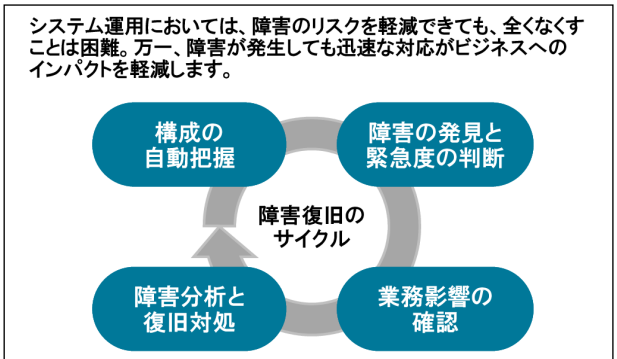

図 17　障害復旧への対応

まずは、構成を自動把握することが大事です。運用業務では、当然ながら、システム全体を見渡す必要があります。ネットワークとそのインフラ回り、インフラリソース、それからアプリケーション、さらにはサービスまでを 3 つのレイヤーに分けて、業務システムの健全性を重要度別に一画面で確認できるようにします（図 18）。

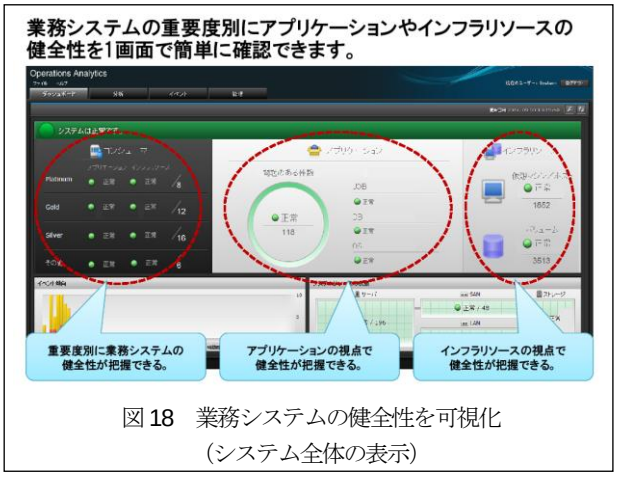

図 18　業務システムの健全性を可視化
（システム全体の表示）

これは、色々な画面を追っていたのでは見過ごすことがあるので、健全性をいつも見られるようにしておくということです。システム運用管理者にとって、何も問題が起きていないなのが一番ありがたい話であって、画面全てに緑色が表示されていれば、現時点で全てのシステムが健全であることがわかります。

図 18 の健全性の可視化の画面例では、アプリケーションの JOB が実際に仮想サーバのどこで動いているのか、サーバが SAN/LAN どのネットワークでどのストレージとつながっているのか、最後にストレージの状態はどうなのかが確認できます。青色のハイライトになっている業務の実行基盤を見ると、これを起点として何が絡んでいるのかわかるようになっています。

図 19 に示すように、重要度プラチナの○○○システムでは 10 の業務アプリケーションが動作していて、アプリケーション AJS-Agent102 は仮想マシン ESX001 の VM103 上で動いているのがわかります。

図 19　業務システムの健全性を可視化
（システムごとの表示）

さらに細部を覗いて見てみると、CPU、メモリ、NIC、HBA、ディスクが水色の丸で囲われているのがわかると思います。これらは決して人手でハイライトをつけているのではなく、あらかじめ構成情報を収集することで、どれとどれがつながっているのか、自動的にわかるようになっているからです。それゆえ、通常はどのシステムがどの仮想マシン上で動いていて、どのネットワークにつながっているのか、ここではストレージは示されていませんが、システムの状態が把握できます。

システム構成が変更されても、最新の構成と接続性を把握できるのが特徴です。どこまで自動で行うのかといった問題はありますが、自動的に構成が変更されることを前提に、最新の状況がわかるということです。

従来の物理環境での管理の場合、設計書に基づいてそれぞれのシステム構成図をつくりますが、構成変更に図面のメンテナンスが追いつかなくて、今の構成と異なっているのは現場では実はよくあることです。仮想環境の場合、構成変更がもっと頻繁に起きるわけで、自動的に情報収集することで、正しいシステム構成の把握を可能にしています。

障害が発生した場合は、あちらこちらの部門の人が一つの部屋に集められて、関係がありそうな事象をホワイトボードに書き出して、ああでもないこうでもないと議論を重ねて、ようやくこの辺りに問題があるかもしれないとなって、そこから真の原因究明が始まることになります。初動作業にものすごく時間がかかっているのが現状です。ここでは、運用管理者とインフラの業務管理者という 2 つの視点から、障害発生時の対処についてお話します。

運用管理者は、まずはインフラのリソースとアプリケーション間のアラートに着目をして障害の切り分けをします。

問題が起きて、最初にジョブで警告が発生したとします（図 20）。ジョブには何か問題があるようだが、それがつながるサーバには何も問題が起きていないようで、他に原因があるということになります。DB のところに赤くマークがついていますが、水色の丸でこのジョブに関係している DB であることがわかります。DB がしきい値を超えたというエラーが出ています。

そこで、ここを起点にして該当箇所の詳細を確認してみると、ログが色々と出ていますが、その中から、DB キャッシュのヒット率が落ちて、ジョブの終了遅延が起きているのが読み取れます。ジョブサーバ上には問題はなく、どれも正常に動いていて、実際にジョブ遅延を起こしたのは DB の方であったことがわかります。この色づけと関連性によって、容易に障害の 1 次切り分けができるようになり、リソースを見ている運用管理者の視点での見直しができるわけです。

図 20　障害発生時の対処を迅速化（運用管理者）

一方で、業務管理者は、業務実行基盤の画面を見ています（図 21）。問題が起きているということで、遅延の発生した時間帯の稼働状況は赤く表示されています。そこで、リソース管理をしている画面を呼び出してみると、グラフ表示で少し色濃くなっているところ、ジョブの遅延、業務の遅延が起きた時間帯に色がつくようになっています。そこを見てみると、ユーザの CPU 利用率が一気に上がっているところがあります。これが今回の障害を起こした要因ではないかと当たりをつけることができます。

次に、このサーバ上で何ゆえ CPU 利用率が上がったのかと考えて、運用管理者に調べてみることを投げかけます。業務管理者は、動作中のサーバのリソースまで調査するわけではないです。それで、先ほどの DB キャッシュのヒット率が落ちて、CPU 利用率が上がって、ジョブの遅延に影響を及ぼしているのではないか、というところにつながって行くのです。

今までなら、ホワイトボードで問題の洗い出しをして、これは大丈夫、これも大丈夫とつぶしていって、やっとのことで当たりをつけています。障害が起きたときには、とにかく初動が大事であって、いかにして早く要因にたどり着けるかにかかっています。運用管理の自動化は、まだ完全にはできていませんが、色々な形で皆様方のご支援に値するものと思っています。

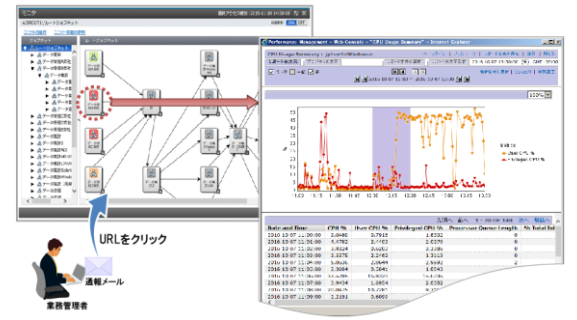

図 21　障害発生時の対処を迅速化（業務管理者）

原因がわかったとして、復旧対処の作業手順や、仮想マシンのマイグレーション依頼など必要となるアクションをあらかじめ登録しておけば、そこから先は確実な対処依頼がいとも簡単に行えます。

先ほどの CPU 利用率が増えているケースでは、少しリソースを増やそうということになります。仮想マシンでリソースを増やすとなると、マイグレーション依頼となります。オペレーターに作業指示を出す上では、漏れがないようにすることが重要になってきます。

復旧の対処には、自動化できるところとできないところがあります。システム障害時の対策として、仮想サーバのプロビジョニング作業などを自動化することで、復旧作業の効率化が図れます（図 22）。

先ほど、オペレーターに仮想マシンのマイグレーション依頼する話をしましたが、指示を受けたオペレーターは、もちろん自分ら作業することも可能ですが、あらかじめやるべきことのシナリオをつくって登録しておけば、実行ボタンを押すだけで完全に自動化できます。例えば、障害発生時にログを出力するサービスとか、仮想サーバのプロビジョニング作業するサービスとかが自動化できます。

そうは言っても、全てを自動化できないところが運用管理の泣きどころです。運用の熟練者に頼らなければいけないところはたくさんあります。しかし、いつまでも彼らに頼っていられないので、自動化できない部分の標準化にも取り組んでいく必要があります。

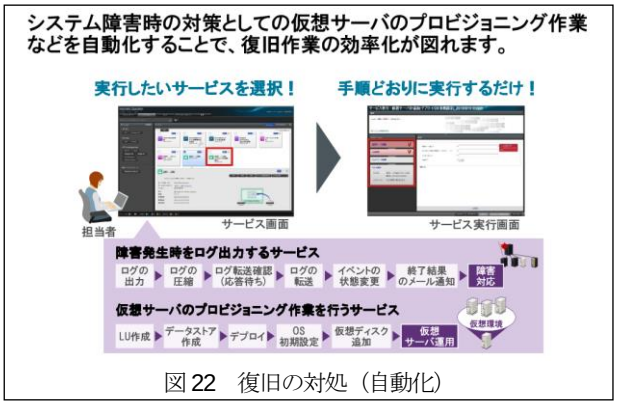

図 22　復旧の対処（自動化）

そこで、運用熟練者の作業手順、業務ノウハウを可視化して、共有して、的確にナビゲートすることで、障害復旧作業を確実に、簡単に進められるようにします。それには、従来のドキュメントと違って、見ながらチェックできるマニュアルのようなものをつくります。

例えば、絶対にやらなければいけない作業に関しては、実際の作業手順を細かく確認しながら操作するということで、チェックボックスを設けて、チェックしないと次の作業ができないようにします。熟練者は、このようなものを見なくてもスムーズに行えますが、いつまでも彼らに頼っていられないので、いかに初心者であっても、このようなナビゲーションツールを使って、あたかも熟練者のように振る舞うことができます。

実際に行ったオペレーションのログも、自動的に出力することができるので、きちんと間違いなく行ったというエビデンスも残せます。図 23 は、運用手順書＋そのナビゲーションのご紹介だけですが、さらに、操作ログの延長で、どの作業で引っかかったとか、どこの部分に時間が掛かったとかの統計を取ることもできます。

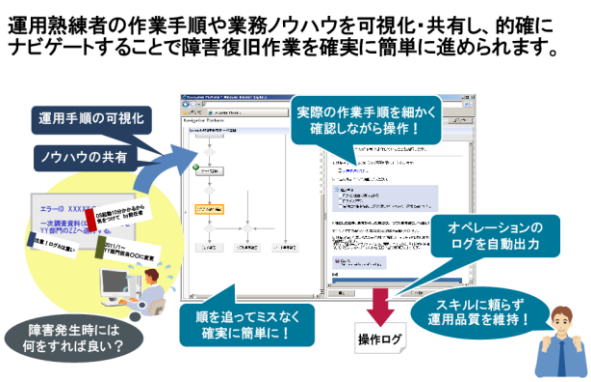

図 23　復旧の対処（自動化できない部分標準化）

ナビゲーションツールに書き表されていない、熟練者だけのノウハウが存在しているのはよくあることです。同じように何か入力するにしても、十人十色で、入力の仕方は結構違うものです。そこで、最も素早く正確に作業している人がどのような入力の仕方をしているのか統計を取って、データ分析し、この人のやり方を標準的な手順書に加えれば、さらに速く、より間違いなく作業できる手順が確立できます。

たぶん、将来的には AI を使ってという話になってくると思いますが、最初に手順書をつくって、それで終わりではなく、さらに改良を加えていくことが必要で、そのときにこの操作ログが役に立ちます。さらに細かく実施していくと、今は人手をかけているところに機械的にやれる作業が生じてくるもので、自動化できる範囲を広げていくためにも、標準化していくことはとても重要であると思っています。

この新しい IT 運用のスタイルは、まだ完成していないところもありますが、IT 基盤の構成把握の自動化、障害発生時の適切な初動によって、まずは障害復旧時間の短縮を図って、障害復旧のサイクルをいかにして早く回すかを考えています。

ビジネス環境の変化に対応して、進化したプラットフォームにおいても高い可用性を持って、万が一障害が発生してもビジネスインパクトを極小化して、ビジネスを継続するという形で、IT 部門の方たちやシステム運用されている方たちにも、新しいビジネス価値を創出することに是非とも携わっていただきたいと思っている次第です（図 24）。

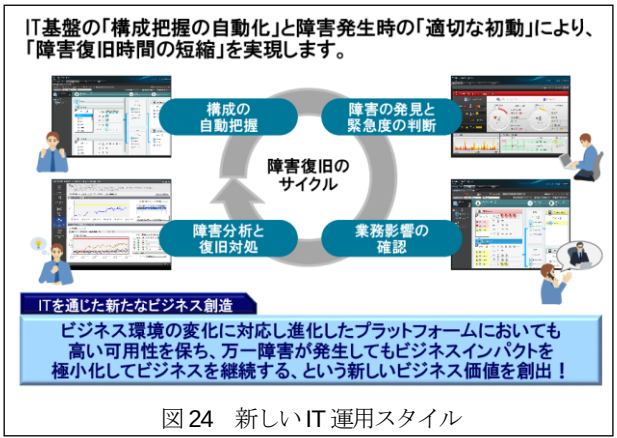

図 24　新しい IT 運用スタイル

日立が考えるＩＴ運用の将来像

ここから先は、現在、日立が考えているIT運用の将来像について、ご説明をさせていただきたいと思います。要素技術に関しては、運用業務という視点から、それぞれの技術がどのように使われているかを見ていただきたいと思います。

本日のこの場での説明は、運用の自動化を行う上で今やるべきこと、今検討しておいた方がよいと思われるレベルのものであって、最終的に具現化されるものは形が変わってしまっているかもしれないことはご承知おきください。

図 25 は、日立が考えている運用管理の自動化と自律化の製品ロードマップです。

最初は、現有の自動化のフレームワークをベースに、まずは各層での自動化サービス群を提供するものです。次は、ルールベースの自律化です。ここからは、自動化から1段階上にレベルアップして、自律化というキーワードになっていますが、ルールベースでの監視や分析のフレームワークをベースにして、各層での分析サービス群を提供するものです。その次は、機械学習などの技術を適用して、運用オペレーションレス化を実現したいということで、現在検討しているところです。

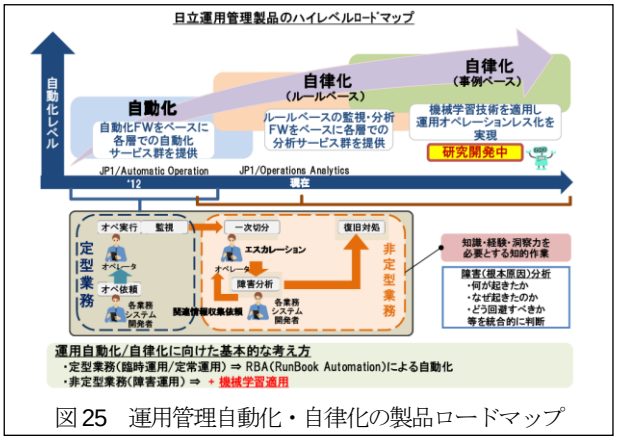

図 25　運用管理自動化・自律化の製品ロードマップ

この運用の自動化、自律化に向けた基本的な考え方を整理すると、運用にかかわる業務は大きく2つ、定型業務と非定型業務に分けられます。

定型業務には、臨時運用と定常運用があります。定常運用の定型業務は、普通の業務なのでわかり易いと思います。臨時運用の定型業務、例えば、メンテナンス業務は、定常的には行ってはいませんが、やるべき作業は決まっているので、一種の定型業務であると考えられます。こういった業務に関しては、ランブックオートメーション技術を使って自動化することを考えていきたいです。

一方で、非定型業務は、知識とか経験とかの洞察力を必要とし、まさしくインテリジェンスな作業です。例えば、障害運用では、障害の根本原因を分析する上で、何が起きたのかを当然見ますが、なぜ起きたのか、どのようにして回避すべきかは、統合的に判断していく必要があります。それゆえ、機械学習を適用していくことになります。ここがまさしく、インテリジェンスと言われている業務にかかわるものと考えています。

自動化、自律化に対して、機械学習を適用していく上での全体像ということで、障害対応の検知、分析、対処のオペレーション最適化に向けた機械学習技術の適用例を挙げます（図 26）。

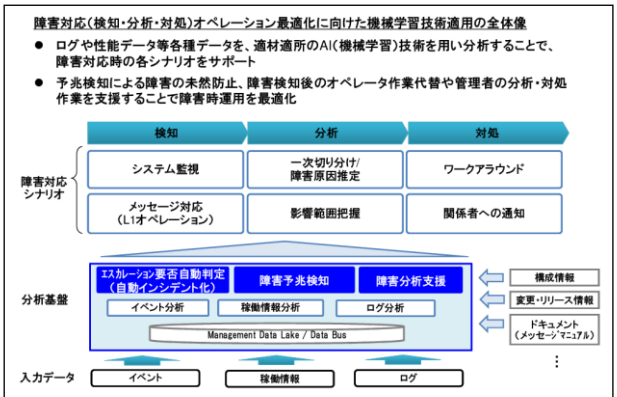

図 26　自動化・自律化機械学習適用の全体像

障害対応においては、ログや性能データ等の各種データを適材適所に機械学習を用いて分析することで、障害対応時の各シナリオをサポートします。一方で、障害が起きる前には、予兆検知で障害を未然に防止したり、障害検知後のオペレーターの作業を代替したり、管理者の分析・対処の作業を支援したりすることで障害運用を最適化します。

このような障害対応のシナリオとしては、検知では、常にメッセージを見ている方たちによるシステム監視であったり、メッセージ対応であったりして、次の分析では、それを基にした1次切り分けであったり、影響範囲の把握であったりして、その次は対処ということで、ワークアラウンドであったり、関係者への通知であったりと、さまざまなことが出現してきます。

このシナリオは、動的に変わってくるイベント、稼働情報、ログなどの入力データと、構成情報、変更リリース情報、ドキュメントなどのあらかじめ与えられる情報の異なる2つを合わせ持つことで、分析基盤の中で色々と分析した結果からつくり出すことになります。この分析基盤にて、AI技術を使って分析していくことになります。

人の動きという意味では、ITサービスマネジメント、ITオペレーションマネジメントの両側面から自動化、自律化を進めていくアプローチが重要になってくると思います。

図 27 は、自律化、自動化に向けて、機械学習の適用範囲の研究開発内容をまとめたものですが、今はまだ研究開発のフェーズです。本日はこの中で、「エスカレーション要否自動判定」、「メッセージに基づく予兆検知」、「システム性能障害の予兆検知」、それから「オペレーション実行条件の自動判定」のところをご紹介させていただきたいと思います。

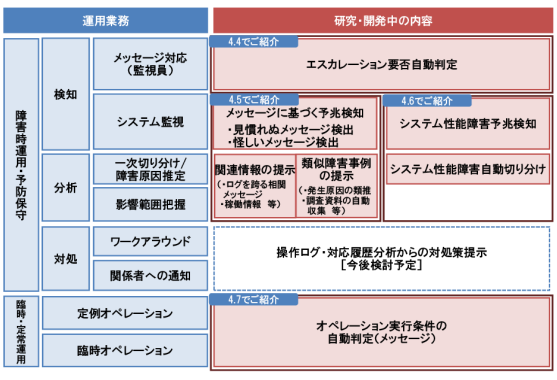

図 27　自動化・自律化機械学習適用の研究・開発内容

エスカレーションの要否判定は、手順書によるエスカレーションの要否判定、そのメッセージ対応作業を代替する技術を研究している段階です。ここはどうしても人手を介することになります。要するに、人が対応する件数を減らすことで、オペレーションの最適化を図っていくものです（図28）。

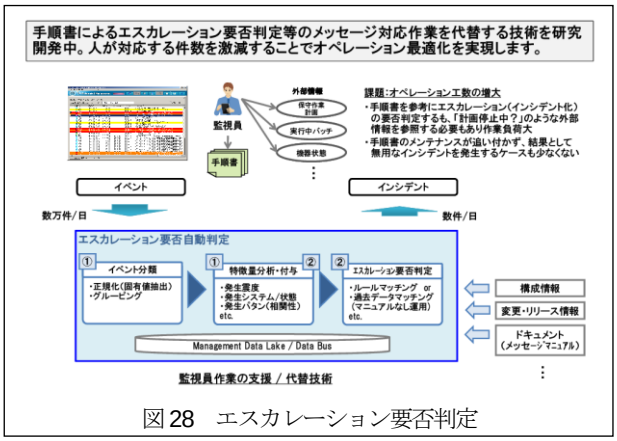

図28　エスカレーション要否判定

なぜ、このような話が出てくるのかというと、オペレーションの工数が増大しているという問題があるのです。手順書を参考にしてエスカレーションの要否を判定しますが、計画停止中のような場合、例えば、臨時運用が行われていてエスカレーションしなくてもよいような場合、これが実際の手順書の中に示されていなくて､外部の情報を参照しなければいけなくなって、作業負荷を増やしている原因となっています。作業手順書に全てのものが書かれていて、それで判断できれば、オペレーターはエスカレーションしなくても直ちに判断ができるのですが、外部要因があるために、自分で判断してよいのか、上位者に上げなければいけないかの判断が、とても難しくなっていることにあります。

もう一つの問題は、手順書のメンテナンスそのものが追いつかず、結果として無用なインシデントが発生しているケースが多いということです。現場ではこれは本当に切実な問題です。システムが正常に動いていれば、手順書のメンテナンスをしなくても何とかなるだろうし、熟練者なら、不完全な手順書を使っても何とかなりますが、新人や不慣れなオペレーターは、複雑化したシステムの中で何かが起きると、あたふたしてしまうことになります。

このような作業を代替していくために、正規化・グルーピングでイベントを分類して、特徴量を分析・付与して、最終的にはルールマッチングでエスカレーションの要否を判定します。これらの処理を自動化して、人手を介さずにできるようにするために、機械学習の適用を考えています。

メッセージに基づく予兆検知は、類似事例を提示するというものです。見なれないメッセージ、障害が疑われるようなメッセージを自動検出することで、発生状況が類似するインシデントを自動検索することを考えています。これは、予兆検知から原因調査の流れを効率化して、最終的には原因究明までの時間の短縮を図るための技術です。

イベント分類で見なれないメッセージやログ分析で怪しいメッセージが出てきたら、過去の事例を事前学習することで、時間的／空間的に近いメッセージや稼働情報などの関連情報を抽出します。さらに、類似事例を提示することで、一つのメッセージから、これから先に何が起きるのかを予兆検知します（図29）。

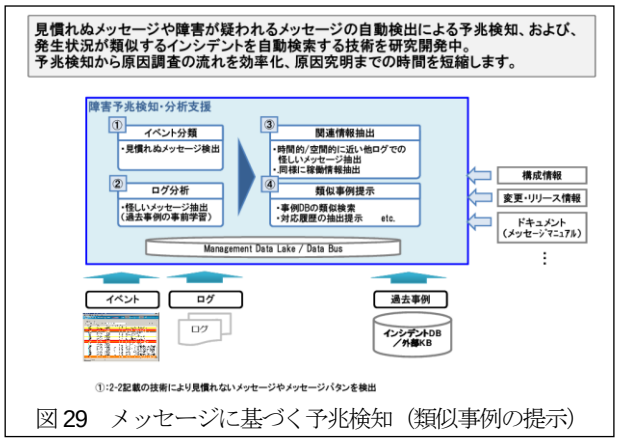

図29　メッセージに基づく予兆検知（類似事例の提示）

システム性能障害の予兆検知からの自動切り分けというのは、障害データを色々と集めると、障害原因に関連する複数のメトリクスの相関性がわかるので、相関性の外れたものからシステムの性能障害の予兆を検知すると同時に、障害原因を推定するというものです（図30）。

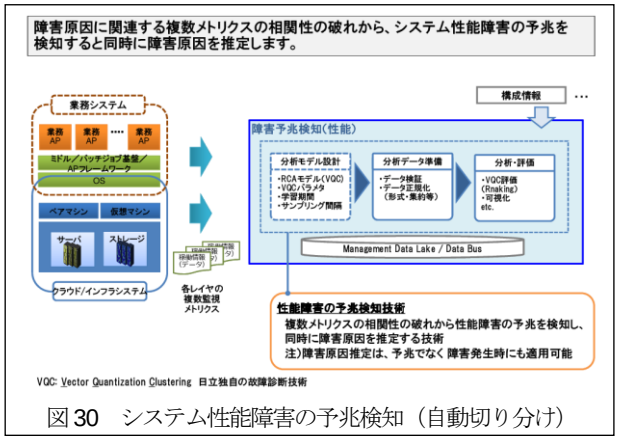

図30　システム性能障害の予兆検知（自動切り分け）

さまざまな業務システムから、各レイヤーの複数監視メトリクスを用意します。それをベースにして、性能にかかわる障害の予兆検知を行います。その分析モデル設計には色々と方法があり、データを準備して、分析・評価を行います。

実際のところ、システム障害はそれほど頻発するわけではありません。正常なログは山ほどありますが、障害ログを集めるのは結構大変なことなのです。もちろん、障害ログがたくさん集まってくるようでは困ります。何かシステム障害が起きて再現させようとしても、ほとんどの場合再現しないです。なぜなら、タイミングやほんのわずかなズレで状況が変わってしまうからなのです。そのため、メーカー側が同じ状況を再現しようとしても、できないことが多いのです

そこで、性能障害の予兆検知をする上で、いかにして障害が起きたデータを集めてくるかが重要になってきます。これを各社バラバラでやっていたのでは、何十年かかったとしても集められません。第三者的なサービスとして、皆で集めてくるようにでもしない限り、AIを使って分析できるほどのデータが集まりません。7～8割方はお蔵入りで、二度と同じ現象が起きないまま、終わってしまうことになると思います。あきらめてしまうというのも選択肢の一つかもしれませんが、やはり、何かが

起きたら原因分析するというアプローチを考えなければいけないと思います。

障害原因の推定は、予兆だけではなく、実際に障害が起きたときにも適用できるので、適用範囲をいかにして広げていくかは、当然考えていく必要があると思っています。

最後は、オペレーション実行条件の自動判定です。定形作業の自動実行は、大概はできると思います。一番やり易くて手をつけ易いところだと思いますが、自動化作業がどれだけ進んでも、作業前後の進行状況確認は人手でしているというのが実状です。

そうなると、作業に伴うイベントの発生パターンを学習させて、オペレーターが確認している作業を代行することで、定形作業前後の人がしている部分をうまく軽減できないかという話になります。作業負担の軽減はもちろんですが、残念ながら、人はミスをしてしまうもので、それを減らせないかと考えているところです。

色々なアプローチ方法があると思いますが、発生パターンを検出して、イベントパターンとマッチングさせていくようなアプローチをしているところです（図 31）。

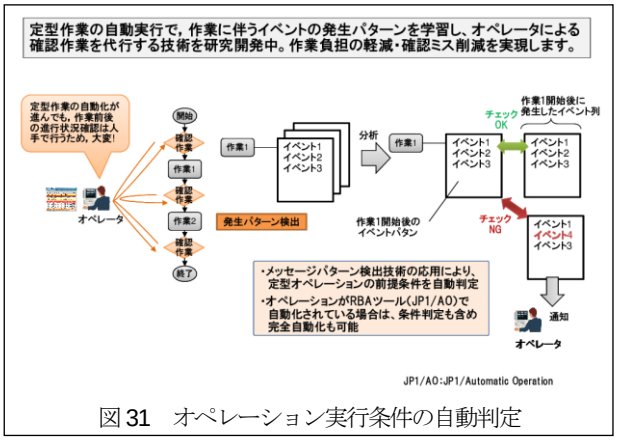

図 31　オペレーション実行条件の自動判定

まとめ

駆け足になってしまいましたが、現状の課題と、日立が現在取り組んでいることのお話をさせていただきました。全てが当てはまるとは思っていませんが、今後、皆様方がご検討される折に、何がしかのきっかけにでもなれば幸いですと思って、お話をさせていただきました。

これで私のお話は終わらせていただきたいと思います。

どうもありがとうございました。

本講演録は、平成 29 年 6 月 9 日に開催されたＳＣＡＴ主催「第 101 回テレコム技術情報セミナー」のテーマ、「AI 等最新技術によるシステム運用の自動化について」の講演内容です。