

Connected Worldの実現に向けて

— オンデマンドでのデータ変換により世界中のデータを利用可能にする —

センサー技術とネットワーク技術の進化により自動車やスマートフォンなどの移動体からさまざまなデータを取得できるようになりつつあります。それらのデータを利用した家庭や企業の電力使用量の最適化や、通行量の最適化を図るアプリケーション、自動車保険やカー・シェアリングなど、他業種間での連携サービスが提案されています。しかし、現在はまだ、センサーなどから取得されるデータは一次オーナーの閉じた世界にとどまるケースが多く、上位の最適化システムや他業種間の連携などに十分に利用されているとはいえません。この潜在的なビッグデータを活用し、企業のビジネスに役立てるためのITの仕組みについて紹介いたします。

① 点在するデータ、つながらないシステム

昨年の東日本大震災の発生後に限られた期間ではありますが、複数の自動車会社などから収集したプローブ・データ（車の運転状況や故障に関するさまざまな情報）と自治体からの道路規制情報などを統合して、特定非営利活動法人 ITS Japan や国土院の地図上に通行実績を表示するサービスが提供されました [1]。この官民連携によるプロジェクトでは従来よりも非常に精度の高い道路規制情報を示すことができ、災害時の物資の輸送に大変役に立ちました。しかし、震災発生後からサービス開始までにかかなりの期間を要し、各社からのプローブ・データの統合には約1週間、官民共同での統合データ・サービスの提供にはさらに3週間程度かかりました。非常時における災害ネットワークを速やかに立ち上げた1つの成功例といえますが、災害後に設計を開始したため肝心の初動には対応できず、また災害時のみに使われるインフラに対する事前の投資も難しいのが現実です。このため、さまざまな領域に点在するデータを統合し、有効に利用する仕組みを通常時に整備しておく必要があります。では、点在するデータの利用状況はどのようなになっているのでしょうか。

家庭における消費電力を最適化する HEMS (Home Energy Management System) ではさまざまな家電や照明器具などのノードが電力の消費量を測定し、その結果を画面に表示します。それらのノードに外部からの制御を受け付ける機能がある場合は、エアコンの温度調整、照明の照度の調整などの必要な処理を行います。各ノードからの情報取得方法の業界標準には、電力線で双方向通信を実現する LonWorks [2] や、日本の電力業界・家電業界で策定した ECHONET [3]、欧州で家庭・ビル管理のために策定された KNX [4] など、さまざまなものがあり

ます。そのため、ホーム・ゲートウェイ（あるいはコントローラー）と呼ばれるボックスには、データを取得するためにさまざまなアクセス・プロトコルを実装しておく必要があります。ビル管理においては同様に BEMS (Building Energy Management System) があります。家庭の場合と同様に、管理の対象となる照明などの機器を束ねるビルディング・コントローラーが系統ごとあるいはロケーションごとに複数存在し、それらのビルディング・コントローラーを束ねる管理システムが存在するというのが一般的な構成です。それでは、家庭用やビル管理用のゲートウェイやコントローラーで取得されたデータは上位のシステムでどのように利用されるのでしょうか。BEMS の場合には BACnet (Building Automation and Control Network) [5] と呼ばれる、一般的なビルに設置されるさまざまな機器の情報にアクセスするための標準があり、これを管理システムとビルディング・コントローラーに実装することにより、さまざまな機器からのデータを集中管理できるようになります。一方、HEMS の場合は、家庭内での消費電力の“見える化”に特化したものが多く、その上位アプリケーション（例えば、地域レベルの電力最適化）とのインターフェースは個別に対応しているものが一般的です。そのため、多くのデータが家庭の中に埋もれてしまっているのです。

自動車では、エンジンやブレーキなどを制御するさまざまな ECU (Electronic Control Unit) が CAN (Controller Area Network) という車両制御用のネットワークに接続されており、それぞれの機器の制御や情報の伝達が行われています。また、それ以外にもさまざまなセンサーが実装されている場合が多く、自動車1台が走ると、環境データ（路面状況、渋滞状況など）、車両データ（走行距離、速度、バッテリー残量など）および運転に関するデータ（ハンドルの切り方、ブレーキのかけ方、アクセルの踏み方など）が

取得できます。それらは非常に価値のあるデータですが、多くの場合には自動車会社内で閉じてしまい、広く一般的に利用される状況にはなっていません。また、近年のスマートフォンのハードウェアの進化により、GPS（全地球測位システム）、ジャイロスコープ、加速度センサーなどが実装され、位置情報・傾き・速度などがスマートフォンで取得できるようになっています。この機能を利用し、スマートフォンを自動車などの移動体に固定することにより、移動体のさまざまな情報を取得することができます。例えば、自動車の走行ルートのみならず、速度や路面の傾きまで取得することが可能です。すでにさまざまなGPSロガー、GPSトラッカーと呼ばれるスマートフォン用のアプリケーションが提供されており、これらを利用、あるいは開発することにより、移動体からさまざまなデータを取得する

ことができます。ところが、このようなアプリケーションはサイロのようにある事業者のある特定のシナリオに閉じている場合が主で、広く業界をまたがってデータを再利用する方向で設計されているものはほとんどありません。

それでは、複数の事業者がデータを参照し合う例にはどのようなものがあるのでしょうか。消費電力の最適化を考えた場合、まず各家庭レベルでの最適化があり、次に地域レベルでの最適化があり、という形で階層的な管理形態が考えられます。また、バッテリーを内蔵している電気自動車（Electric Vehicle：以下、EV）が普及してくると、家庭内では電力会社から供給される電力の利用と、車載バッテリーへの蓄電と、充電された車載バッテリーの家庭での利用との最適化が考慮されるようになります。バッテリーを積んだEVは電力のポータビリティを実現し、EVのバッテリーを利用することで、災害時の非常電源の確保が可能となり、都市レベルでの電源利用の最適化に貢献します。このようにさまざまなレベルでの最適化が望まれますが、現実的には、データの提供側とデータの利用側を積極的に結び付ける共通のデータ・フォーマットやアクセス・プロトコルの定義には定義には時間がかかることと、データの提供者と利用者をつなげるブローカリング（仲介）事業自体が成立していないこともあり、個別に対応する形での地域レベルの消費電力最適化の実験実施にとどまっているのが現状です（図1）。

また、1つの管理対象を複数の異なった上位アプリケー

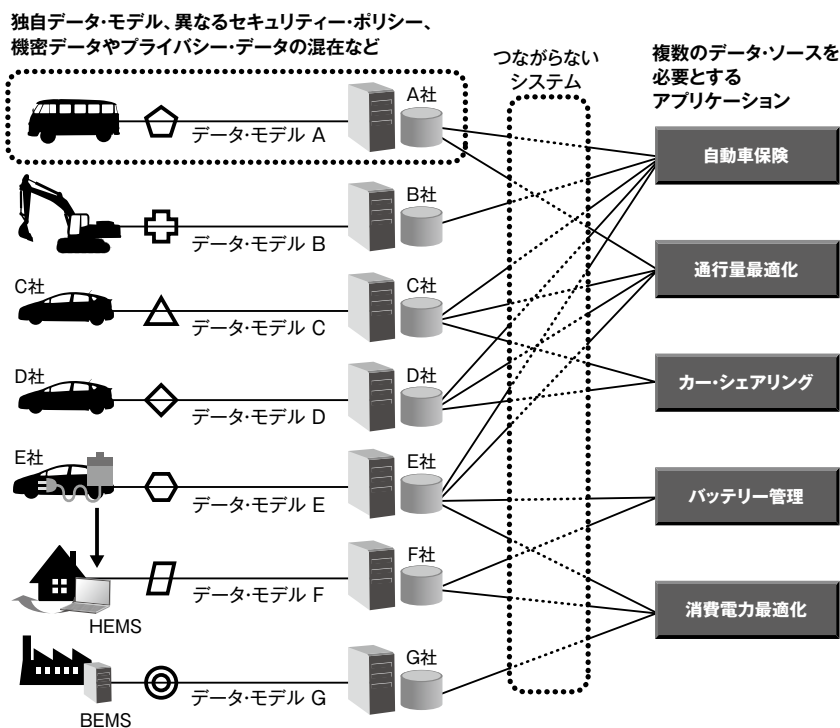


図1. つながらないシステム

ションが管理することが多々あります。EVを例にとると、「位置情報を利用した都市部におけるトラフィック全体の最適化を行うアプリケーション」「バッテリーの残量を確認し走行ルートを探るアプリケーション」「急速充電スタンドにおける待ち時間を最小にし、目的地まで最短時間で到着するプランを提示するアプリケーション」などが考えられます。しかし、それぞれがサイロのように独立している場合には、各アプリケーションからの指示内容が矛盾してしまう場合もあります。

このように、わたしたちの身の回りの機器にはさまざまなセンサーが実装されており潜在的なビッグデータが点在していますが、その利用方法は成熟しているとはいえません。しかし、データの連携、アプリケーションの連携は System of Systems を実現する Smarter Planet [6] には不可欠です。

② 潜在的なビッグデータをビジネスに結び付けるには

高度にインテグレーションされている組み込み機器や自動車の制御システムから収集されるデータのフォーマットは独自仕様であることが多く、それらを利用したアプリケーションの開発には個別対応が必要となります。アプリケーション開発者からすると複数のベンダーをサポートするためにはその数だけ個別対応が必要となり開発コストがかさむことになります。この問題を解決するためにはその業界で標準フォー

マツを策定する方法が一般的ですが、多くの場合、標準フォーマットについて関連する複数の企業間で合意するまでにかかりの時間を要してしまい、機器の進化に追いつけないという問題を抱えています。また、標準があったとしても、都市運営の効率向上と住民や企業への提供サービスの品質向上を目指すスマート・シティのように、異業種を統括する上位アプリケーションとの連携では、必要なデータがすべての業界の標準で提供されているかどうかは分かりません。このような上位アプリケーションとの連携を考えると、データのオーナー側で上位アプリケーションの要求に応じてデータを選択し提供する方法が有効です。もちろん、業界標準のフォーマットで提供する場合にも、内部データの変換が必要なこともありますし、複数の異なった業種の上位アプリケーション（例えば、交通の最適化を目指す Smarter Traffic と電力の最適化を目指す Smarter Energy）と連携する場合にはそれぞれのフォーマットでデータを提供する必要があります。

一方、上位アプリケーションが必要とするデータは複数の内部データの組み合わせにより作られる場合もあります。例えば、自動車のある特定区間における平均速度を求める場合、上位アプリケーションが独自で行うためには莫大なデータをデータ・オーナーから取得し計算する必要がありますが、データ・オーナー側である程度の処理を行えば、上位アプリケーションの処理は簡単な計算で済みます。従って余分なビッグデータの転送の必要はなく、ネットワーク速度やコストなどの心配もありません。データを提供する側と利用する側でのデータ・フォーマットに関するネゴシエーションによりデータ利用コストを最適化することができるのです。

また、データ・オーナー側のデータには上位アプリケーションでは不必要なデータ、設計に関連する機密データ、計測エラーによる異常値、あるいはプライバシーに関連するデータが混在している場合が多々あります。これらを上位ア

プリケーションに効率よく、また、安全に提供するためにはデータ・オーナー側でのフィルター処理や匿名化処理が必要となります。

データ・オーナー側で、機密データやプライバシーに関連するデータを除きながらオンデマンドで複数フォーマット（データ・モデル）にリアルタイムで内部データを変換する仕組みを導入し、積極的にデータを提供することにより（図2）、容易にアプリケーションを開発できるようになり、他の業種や情報サービスと連携して、通行量最適化や消費電力最適化など業界をまたがる新しいビジネス・アプリケーションを作ることが可能になるのです。

③ ビッグデータ利用における課題点

前章で述べましたように、潜在的なビッグデータをビジネスにつなげていくためには、点在しているビッグデータを上位アプリケーションが利用できるような変換していく必要がありますが、実際にビッグデータを利用する際にはさまざまな課題点があります。

システムが扱うデータには構造化データ、非構造化データ、頻繁に更新されるデータ、ほとんど更新されないデータなど、さまざまなデータがあり、データ処理のみならずデータ利用に関するBSS（Business Support System）との連携も考慮する必要があります。ビッグデータを利用するシステムは、個々のミドルウェアのみで完結するものではないので、全体のデータ・フローを意識し、システム全体のセキュリティー（データ暗号化、経路保全、認証認可）を考慮してシステムを構築しなくてはなりません。また、ミドルウェアが多様になることでシステム構築・管理に必要とされるスキルの種類が多くなることも考慮してシステムを設計する必要があります。

センサーなどで取得するデータ項目の増加や通信技術の

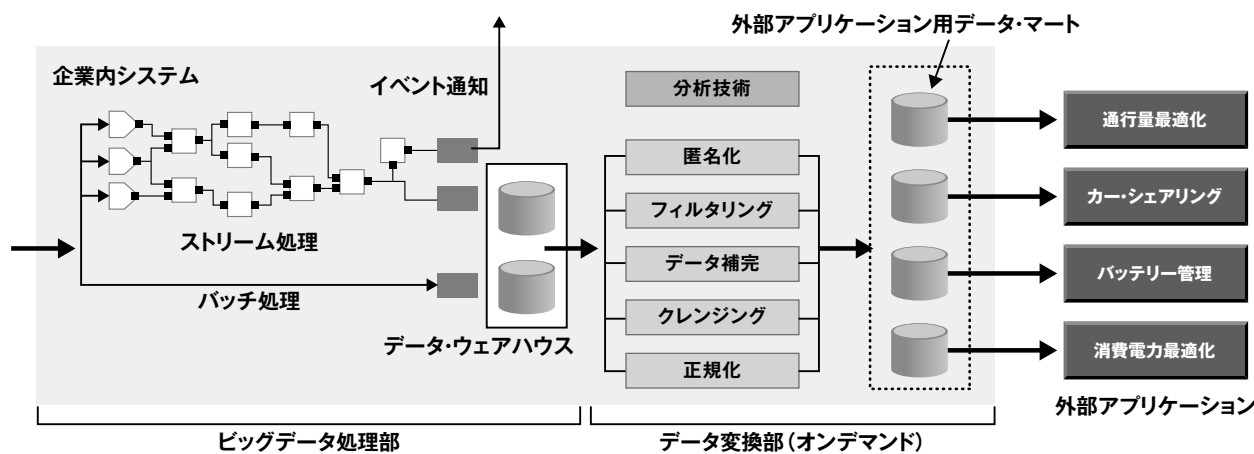


図2. ビッグデータをビジネスにつなげるシステム

進化により、また対象となる移動体の増加や測定頻度の増加により、取得されるデータ量が飛躍的に大きくなることも意識して拡張可能なシステムにする必要があります。ただし、システムの観点では、スケールアウト拡張システムがほとんどであり、ノード数は増大傾向にあります。

一方、システム・テストの観点では、テスト・データの準備には膨大な負荷がかかることを考慮する必要があります。ビッグデータといえる実データを入手できればいいのですが、テスト・データを独自に作成するケースが多く、「Big」といえるサイズまでデータを収集・作成するには非常に多くの困難を伴うことが多いのです。

4 ビッグデータ処理システムの構成例

前章で挙げた課題は実システムで1つ1つ解決していく必要がありますが、一例としてここではIBMの試行プロジェクトとして実際にシステムを構築したケースを紹介します。

IBMではSmarter Planetの一環として、「つながるクルマ」をテーマに次世代のモビリティを実現する基盤、SECA-V2Xを提唱しています。第1章で挙げたように、点在するデータは結び付けてこそ価値を持ちます。SECA-V2Xは世界中を走る自動車などのモビリティ機器が発信するプローブ・データを集約し、情報価値に換えて提供するためのプラットフォームです。SECA-V2Xが実用化されれば、さまざまな業種にまたがるシステム全体で分析結果などを広く活用し、交通施策やリアルタイム交通渋滞予測などへ適用することが可能になると期待されています。このプラットフォーム実現のためには、前章で述べたようにデータの収集、蓄積、解析、利用でそれぞれのミドルウェアが有

機的に連携し、一連の処理を行うことが必須となります。そこで相互連携を確認するための試行プロジェクトを2011年に行いました。

(1) デザインとアプローチ

システムのアーキテクチャー・デザインに関しては、構築試行前のプロジェクト段階でIBMのアーキテクトによりビジネス・シナリオおよびユース・ケースからの要件の抽出、データ・モデル設計とシステム・コンテキスト定義を行い、大枠のアーキテクチャーが組まれていました。構築プロジェクトではこのアーキテクチャーに基づき、優先度の高いコンポーネントとデータ・フローを選定するという論理モデルの精緻化、使用する技術要素・ミドルウェアの選定、物理設計を行いました。

システムの概要図は図3の通りで、プローブ・データを収集・処理するIBM InfoSphere Streams(以下、Streams)、蓄積を行うIBM DB2(以下、DB2)、解析および可視化を行うApache Hadoop(以下、Hadoop)ならびにその分散ファイル・システムであるHadoop Distributed File System(以下、HDFS)、IBM SPSS Modeler(以下、SPSS Modeler)、IBM Cognos Business Intelligence(以下、Cognos)、各ミドルウェア間でデータの連携を行うIBM WebSphere MQ(以下、WMQ)とIBM InfoSphere DataStage(以下、DataStage)、そして管理アプリケーション用のIBM WebSphere Application Server(WAS)という、多様なコンポーネントが相互連携するシステムになっています。なお、図中の破線部は実装対象外のコンポーネントなので個々の解説は省略します。

(2) 構築と仮想環境上への展開

このシステムの実装環境には、展開のスピードと移行容易性を勘案して社内の仮想環境を選定しました。構築完了後、IBMのクラウド・ソリューション・デモシステム上に仮想マシンを移管しています。試行プロジェクトの段階では相互連携の実証を重視し、パフォーマンスや耐障害性などの非機能要件は考慮対象から外し、各ミドルウェアはいずれも単体構成としました。それでも前述したすべてのミドル

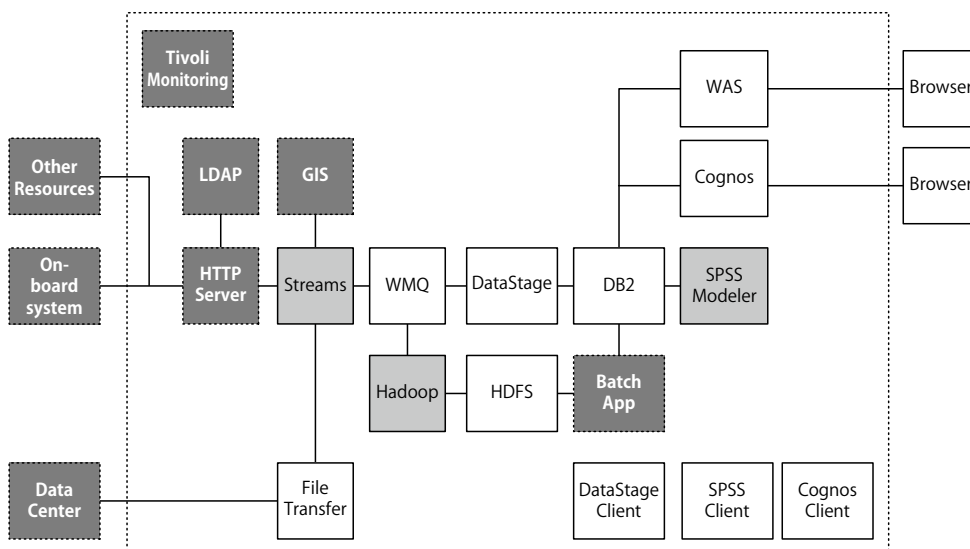


図3. システム概要図

ウェアを構成するために5台のサーバーと1台のクライアント、計6台の仮想ノードが必要となりました。サーバー OS は 64bit Red Hat Enterprise Linux にそろえ、Cognos、SPSS Modeler などの一部のミドルウェアの管理クライアント用に Windows を構成しました。

(3) テスト・データの準備

このシステムでは、車両の走行データを扱いますが、試行プロジェクトでは手元にデータがなかったため、シミュレーターなどで生成するか実際のデータを収集する必要がありました。このときはスマートフォンの GPS トラッカー・アプリケーションを使用して、実際の走行データを社内の有志で一定期間集め、それを利用しました。それでも「ビッグデータ処理基盤」を検証するには十分な量ではなかったため、レコードのコピーなどで増量して対応しました。

(4) データ連携

車両走行情報データがシステムに取り込まれると、解析までに大きく2つの流れを構成します。Streams でデータ・フォーマットの一次変換実施後、(A) DB2 に蓄積し、SPSS Modeler での走行経路の集約、Cognos での車両ごとの速度傾向リストの作成が行われるフローが1つ目、(B) Hadoop の MapReduce 処理 (ビッグデータの分散処理) により特定の地域の通過プロット数の計上を行うフローが2つ目です。2つのフローの分岐は Streams の後段に控える WMQ での Pub/Sub (Publish/Subscribe 処理) により行います。Subscriber の1つは ETL (Extract/Transform/Load) ツールの DataStage であり、DB2 への入れ込み処理を行います。もう1つの Subscriber はシェル・スクリプトで、読み取ったメッセージを HDFS へ配置します。Streams は TCP ポートでリクエストを受領しオンラインで変換処理を行い、Hadoop や SPSS Modeler、Cognos で行われる後半のデータ変換、解析処理はいずれもバッチ・プログラムとして定期インターバルで実行されます。

(5) 状況監視アプリケーション

システムでの処理の進捗状況は、各所でのファイルや RDB、キューなどのデータ量で計測できます。管理者が

SECA-V2X Status and Data Transition Report

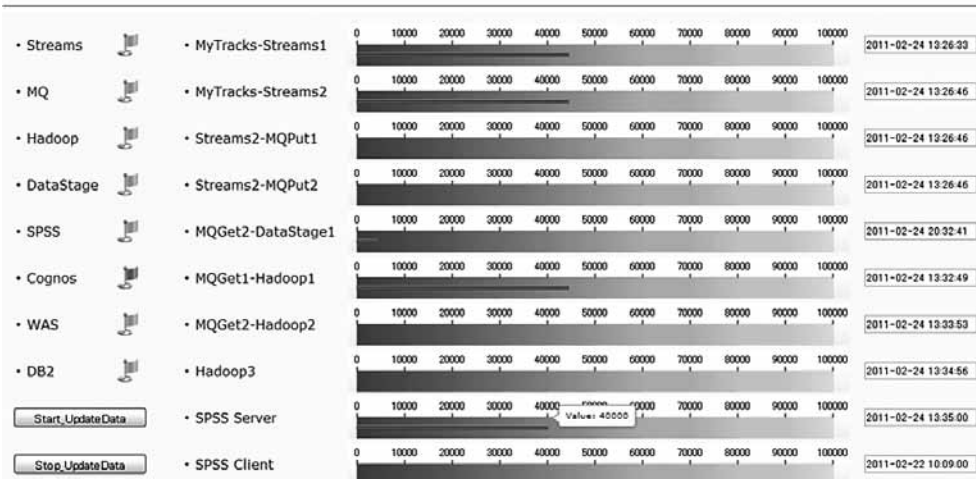


図4. 状況監視アプリケーション

処理状況を把握するために、これらのデータ量を集計して表示する Web アプリケーションも作成しました (図 4)。各処理パートにおける処理状況は数秒の頻度で更新されるため、JavaScript で画面を定期的によりロードし、最新状況が反映されるようになっています。

(6) 考慮点と総括

多種のミドルウェアを連携させ、ビッグデータをフロー処理させることを実現したシステムですが、これまでに挙げたほかにも以下のような考慮点がありました。

- 構築・開発・運用メンバーに多くのスキル・セットが必要
- ノード数増加時のコスト、必要なソフトウェア・ライセンスの増大
- 解析処理内容によりシステム負荷が変動するため、早い段階のキャパシティー・プランニングが困難

ビッグデータを扱うシステムとして、Hadoop や Streams など個別の技術がフォーカスされがちですが、どこで生成されたデータが、どんなフォーマットで蓄積され、どう変換・利用され、アーカイブされていくのか、というトータルデータのフローを把握したデザイン・運用への配慮が重要であるといえます。

5 データセンターからデータ発生源にインテリジェンスをシフトさせる

前章でデータの提供側と利用側を結び付けビッグデータの有効利用を実現するシステムの構築について解説しましたが、ここでは扱うデータの種類に応じた分析技術の違い

について取り上げます。

ビッグデータ処理の方法には蓄積されたデータ (Data at Rest) の分析と連続的に変化するデータ (Data in Motion) の分析があります。近年のセンサー技術や GPS の普及と無線通信技術の進化により、移動体からさまざまなデータをリアルタイムで取得し、ビッグデータを構成するようになりつつあります。このようなケースでは、時間をかけて蓄積されたデータを分析し知見を得ようとするトラディショナルなビジネス・インテリジェンスの機能ではなく、時刻・位置情報とその時のさまざまなセンサー情報をベースとしたリアルタイムの分析が必要となってきます。特に、故障予測や衝突可能性検知といった安全・安心に関するアプリケーションの場合、情報の取得、分析、フィードバックのすべてがリアルタイムで行われなければなりません。バックエンドのデータセンターに一旦情報を集めてから分析を行う手法では遅いのです。また、データの発生源において連続的に無意味なデータを取得し続けることも少なくありません。例えば、正常値を示し続けているセンサー・データなどです。これらをそのままネットワークを通してバックエンドまで送り続けることはコストの観点で当然問題となります。データの発生源でのデータ処理は不可欠になりつつあります。分析技術 (インテリジェンス) をできる限りデータの発生源に近いところに組み込み、リアルタイムの分析を行い対応することで安全・安心が実現でき、大きなビジネス機会の損失を防ぐこともできるのです。

IBM は世界中の基礎研究所でさまざまな分野における分析技術を研究しています。それは自動車、銀行、小売などの業界に特化したものや、データセンター、ネットワーク、組み込み機器などに向けた汎用性が高いものがあり、その総数は 80 以上になります。その中には自動車の故障の予測を行う分析技術もありますし、「ヒヤリ・ハット」の発生形態を分析する技術もあります。このような分析技術をデータの発生源からバックエンドのデータセンターまでの適正な位置 (車載器、ネットワーク、構内システム、大規模データセンターなど) に配置し、前述のビッグデータ処理インフラとオンデマンドでのデータ変換により、点在しているデータをビジネスに結び付けていくことができると考えています。

6 おわりに

本稿では、点在するデータをつなぎ、外部アプリケーションの開発・普及を促進するためのオンデマンドでのデータ変換と、ビッグデータを効率的に処理するための分析技術の組み込みについて述べてきました。これらの仕組みにより、法規制やセキュリティー・プライバシーに関する考え方の異

なる世界中のさまざまな地域において安全にデータを処理しビジネスを迅速に立ち上げていくことができると考えています。IBM では今後もお客様のグローバル展開を支援する技術の研究開発に取り組んでいきます。

[参考文献]

- [1] 国土交通省 奥村康博: 東京モーターショーシンポジウム 2011 発表資料「東日本大震災での取り組み」, <http://www.jari.or.jp/resource/uploads/Symposium2012-01.pdf>
- [2] LonWorks, <http://www.echelon.co.jp/technology/lonworks/default.htm>
- [3] ECHONET, <http://www.echonet.gr.jp>
- [4] KNX, <http://www.knx.org>
- [5] BACnet, <http://www.bacnet.org>
- [6] Smarter Planet, <http://www.ibm.com/innovation/jp/smarterplanet/>

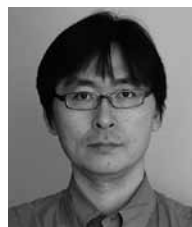


日本アイ・ビー・エム株式会社
ソフトウェア開発研究所
ディステイグイッシュト・エンジニア

浅井 信宏 Nobuhiro Asai

[プロフィール]

1984 年日本 IBM 入社。オペレーティング・システム、ホスト端末、通信、システム管理製品などの設計・開発に従事。1997 年より IBM Austin 研究所に赴任。2000 年より大和ソフトウェア開発研究所においてソフトウェア・アーキテクチャーを担当。近年は、Smarter Planet Appliance や Smarter Mobility のアーキテクチャー設計を担当する。



日本アイ・ビー・エム
システムズ・エンジニアリング株式会社
BAO ソリューション推進
シニア IT スペシャリスト

山口 崇 Takashi Yamaguchi

[プロフィール]

1994 年日本 IBM 入社。IT スペシャリストとして WebSphere Application Server を中心に、ハイ・バリュー製品の技術支援を担当。近年は InfoSphere BigInsights、InfoSphere Streams などビッグデータ関連の技術支援を行っている。