クラウドにおける大量データ処理技術

Big Data Processing in Cloud Environments

● 土屋　哲　● 坂本喜則　● 槌本裕一　● Vivian Lee

あらまし

近年，ICT機器やネットワークの低価格化に伴い，実世界で計測された様々なデータがクラウドに集まり，大量のデータ分析から企業活動や社会に有益な「価値ある情報」を得ることへの期待が高まっている。しかし，本格的なデータ活用では数十Tバイト～数十Pバイトという大量データを扱うこととなり，従来のICTとは異なる技術が必要となっている。また，社会システムなどの重要サービスは24時間365日止めるわけにはいかず，システムを動的に構成変更する技術が求められている。

富士通および富士通研究所では，こうしたクラウドでの大量データ処理の基盤となる技術や応用を促進する技術の開発に取り組んでいる。

本稿では，その一端として，基盤技術である分散データストアと複合イベント処理，およびデータ処理向けワークフロー記述を紹介し，今後のデータ処理技術の方向性を展望する。

Abstract

In recent years, accompanied by lower prices of information and communications technology (ICT) equipment and networks, various items of data gleaned from the real world have come to be accumulated in cloud data centers. There are increasing hopes that analysis of this massive amount of data will provide insight that is valuable to both businesses and society. Since tens of terabytes or tens of petabytes of data, big data, should be handled to make full use of it, there needs to be a new type of technology different from ordinary ICT. Furthermore, as important services such as social infrastructure services should keep running 24 hours a day, 7 days a week, technology to dynamically change system configurations is also required. Fujitsu and Fujitsu Laboratories are working on basic technologies and application-promoting technologies for processing big data in a cloud environment. In this paper, we introduce two fundamental technologies: distributed data store and complex event processing, and workflow description for distributed data processing. We hope this gives a perspective on the direction in which this new field should head.

ま え が き

クラウドコンピューティングの本質の一つは，リソースやデータをインターネット上のデータセンターに集約することにある。現在の各種クラウドサービス（IaaS, PaaS, SaaS）は，アプリケーションの実行環境をサーバ，OS，ミドルウェアなどの様々なレベルで集約して，共用化することで実行効率の向上を実現している。一方，これとは別に，データをクラウドに集約し，クラウドの強力な計算能力で分析を行うことも始まっている。

このように，今，クラウドはアプリケーション集約・共用からデータ集約・活用への拡大期間にあるが，本格的なデータ活用のためには，数十Tバイト～数十Pバイトというデータを扱うことが必要で，従来のICTとは異なる技術が求められている。

本稿では，クラウドでの大量データ処理の基盤技術である分散データストアと複合イベント処理，および欧州富士通研究所でのデータ処理向けワークフロー記述の研究を紹介し，今後のデータ処理技術の発展方向を展望する。

クラウドにおける大量データ処理の全体像

近年，ICT機器やネットワークの低価格化に伴い，実世界で計測された様々なデータがクラウドに集まるようになっている。例えば，携帯電話や自動車に装備された位置センサ（GPS）の情報，あるいは店舗レジの精算記録などは，発生した位置と時刻とともに記録され，ネットワークでデータセンターに転送・蓄積されている。こうしたデータを時系列で分析し，さらに個人の購買行動と結び付けた推測などを行うことで，これまで単なる記録に過ぎなかった「データ」から，個人の購買行動予測などの「価値ある情報」を導出することが可能になり始めている。こうした処理は“Big Data”とも呼ばれており，ある試算によれば，米国で発生するイベントデータは推定で700万件/秒であり，これを圧縮なしで単純に蓄積すると総計で数十～数百Pバイト/月にもなる。[1] この数値は，まだ実際にはデータセンターに転送されて処理されている量ではないが，今後こうした細かなデータが得られることで「来週木曜日に，ある個人がどこでどういう買い物をするか」といった企業や個人にとって「価値ある情報」が得られるという期待が高まっている。

以前より，コンビニエンスストア業界などでは，POSデータの分析から戦略的な出店計画を立案する，といったことは行われていた。しかし，最近注目されているのは，在庫や購買という蓄積された数値データ（ストック）に加えて，常に動いている個人の位置を時系列として見るようなイベントストリーム（フロー）のデータを対象にすること，あるいは，企業活動の最適化ではなく，個人向けに個別の結果を出すことを目標にすることである（**表-1**）。

こうしたクラウドにおける大量データ処理を実現するシステムの構造は，大量のイベントストリームを扱うために，多数のサーバで判定，蓄積，分析を行うサブシステムの組合せとなる（**図-1**）。

実世界の対象物としては携帯電話や自動車が挙げられ，携帯電話なら1億台，自動車なら数千万台という非常に多数となる。またイベント発生の特徴として，人気歌手のコンサートのように，ある時点で突発的に発生したり，季節や一日の中で量が大きく変動したりすることが挙げられる。

そこで，判定処理では，社会システム規模の最大数百万件/秒にも対応でき，またイベントの突発的な発生や急激な変動があっても取りこぼさず，すぐに判定を返せるように，多数のサーバを並列に動作させ，サービスを止めずに，需要に応じてシステム構成を柔軟に変更させることが求められる（図-1①）。

また，分析処理では，数十Tバイト～数十Pバイトものデータの統計分析を短時間で行うために，数百～数千台のサーバによる分散並列処理を行う必要がある（図-1②）。

さらに，データ量が急に増えても蓄積できるように，書込み性能が高く，かつ，容量に応じて柔

表-1　大量データ処理の対象と目的

目的＼対象	ストック（数値，関連性）	フロー（イベント，時系列）
集　団	陳列棚の配置（POSの活用）	スマートグリッドの電力割当
個　別	ECサイトのリコメンデーション	カードの不正検知，行動ターゲティング広告

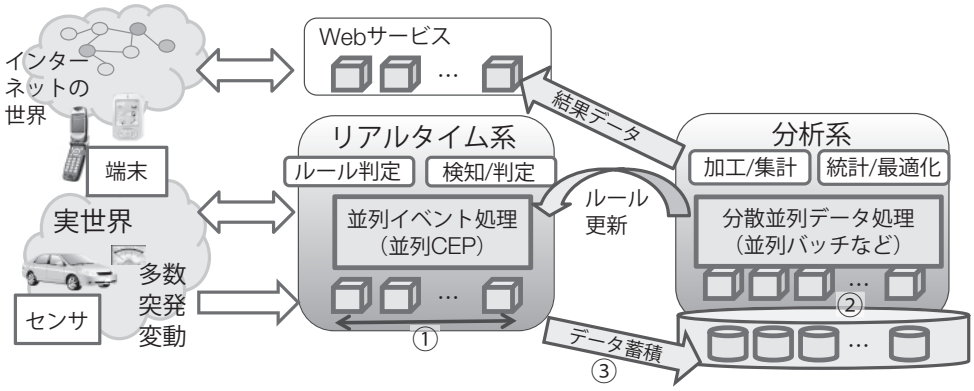


図-1　クラウドにおける大量データ処理の全体像
Fig.1-Overall picture of massive data processing on cloud.

軟に拡張可能である必要があり，数百～数千もの並列で実行される分析処理に対して，効率的にデータを供給する必要がある（図-1③）。

以下の章では，こうした要件に対応する基盤技術の中核を説明する。

分散データストア

クラウドの大量データ処理では，アプリケーションサーバなどデータにアクセスする主体が数百個になり，非常に多数の読み書き要求が発生する。そのため，データ保管側のサーバも数十～数百個用意し，適切に読み書きを負荷分散することや，また，多数のサーバのどれかが故障しても，サービス全体が止まらないことが必要とされる。

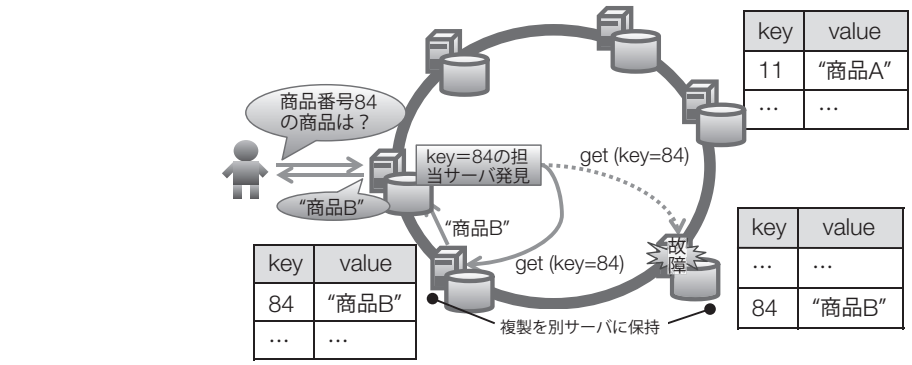
分散データストアの代表例は，分散キーバリューストア（KVS）であり，キーとバリューからなる単純なデータ構造を多数のサーバに分散して保管し，キーで指定した要求に応じていずれかのサーバで読み書きを行って応答を返す方式である（**図-2**）。多数のユーザの読み書き要求が同時に来ても，1台のサーバに負荷が集中しなくなるので性能が高く，さらに，一つのデータを別のサーバに複製して格納することでサーバが1台故障してもデータはなくならず，故障にも強いという特性を持つ。分散KVSは，こうした性質から，多くのユーザのアクセスを処理しなければならないオンラインショッピングなど大規模なWebサイトのセッション情報の保管などによく使われている。センサイベントを取り扱うシステムの場合は，センサデータを記録する部分，あるいは分析した結果をWebアプリケーションで外部に提示する部分など，たくさんの読み書き要求が発生する部分で使われる。

このように，分散KVSは，データを複数サーバで分散して持つことで高い性能・可用性を発揮するが，一方で，分散しているためにいくつか特定の処理は苦手としている。その苦手の代表例が「集計処理」で，複数のデータを集めて結果を出すため，データがサーバ間に分散して存在している分散KVSでは，集計のためにサーバ間の通信が非常に多数発生し，時間がかかって遅くなってしまう{図-2（b)}。

そこで，富士通研究所では，分散KVS上で高速な集計処理を実現する方式の研究開発に取り組み，研究用プロトタイプで従来方式の約8倍の高速化を実現した。[2] 新方式では，分散KVSで効率的に動作する数種類の基本操作（rekey，map，filter，reduceなど）を準備し，集計処理をそれら基本操作の組合せで実現している。それぞれの基本操作は分散KVSで効率的に並列動作するように設計されているため，集計処理全体も分散KVS上で効率的に実行される。

ソーシャルネットワークサービス（SNS）でのリコメンデーション（関連するページの推奨）を想定例とし，ユーザのアクセスログ100万件を集計する実験を行った。サーバ台数を2倍にすることで処理時間を2分の1に削減でき，サーバを追加することでリニアに性能を向上できることが確認できた。

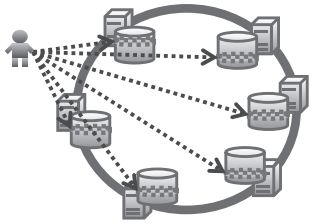
またほかの分散バッチ方式と比較して，約8倍の



（a）分散KVSの概要

```
sum = 0
for key in kvs_a.keys():
    value = kvs_a[key]
    sum += value
return sum
```

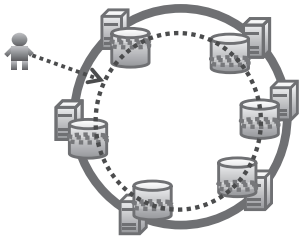
1億個のキーがあれば
1億回通信が発生！
台数を増やしても性能は一定
0.1 ms×1億＝1万秒



（b）従来の分散KVS

```
return kvs_a.get_sum_of_values()
```

分散環境で並列に実行し，
通信を最小限に抑えることで，
台数に比例した性能を実現。



（c）あるべき姿

図-2　分散KVSの概要，および集計処理のあるべき姿
Fig.2-Overview of distributed KVS, and ideal form of aggregation.

性能向上が確認できた（**図-3**）。

複合イベント処理（CEP）の並列化

複合イベント処理（CEP：Complex Event Processing）とは，実世界の活動や業務で常時発生する，複雑で大量のイベントの系列をリアルタイムに処理・分析する技術のことである。このイベント処理技術の利用によって，例えば交通分野においては，時々刻々と変化する個々の車の位置情報を大量に収集し，道路ごとの渋滞状況を分析し，それを基に渋滞回避ルートを提示する応用が考えられる。

今後，クラウドを使った社会向けサービスでは，より大量になっていくイベントに対し，負荷が大きく変動したり，突発的に発生したりするケースにも対応する必要がある。また，災害の予兆検出や防災など重要な社会システムは，24時間365日止めない運用が求められる。そこで，富士通および

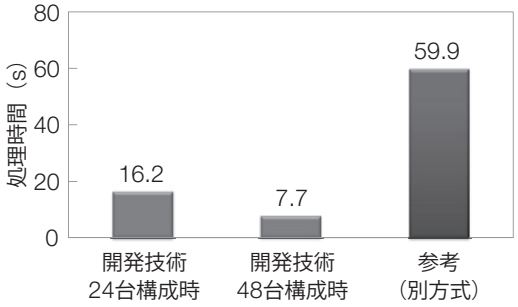


図-3　100万件の集計の測定結果
Fig.3-Aggregate results of measuring 1 million pieces of data.

富士通研究所では，クラウド環境において，イベント処理の動的な負荷分散を行い，サービスを止めずにリアルタイム性を保証する技術が重要であり，他社差異化になる技術と位置付けて，研究開発を進めている。

この技術は，**図-4**にあるような処理モデルにおいて，イベントストリームに対する処理｛データ

操作ルール（クエリ）の集まり}をイベントストリームの特徴や負荷に合わせて動的に負荷分散し，処理を最適化するものである。

CEPは，**図-5**に示す基本構造になっている。

CEPを動的分散するためには次のような技術課題がある。

・イベント処理エンジンの「処理状態（State）」の分割と移動（例えば，イベントの到着順序や演算結果）
・イベントの到着順序の保証や時刻情報の同期

2010年度は，CEPの動的負荷分散の基本動作の試作を行った。要点のみを簡単に記述すると，現用系と切替系を並行動作させ，CEPエンジン内で使用するルールとデータを現用系から切替系へコピーし，かつ並行動作中に届いたイベントは現用系と切替系の両方に流すことで，「処理状態」を合わせる方式を採っている。

また，イベントの到着順序の保証に関しては，実行環境を管理するマネージャが，個々のエージェントに指示を送り，構成変更の起動と終了時に処理の全体で同期を図ることでイベント到着順序の入替りを防いでいる（**図-6**）。

これらの基本動作をプロトタイプとして，人の位置・嗜好情報と店舗の商品情報をマッチングさせて，適切なクーポンを発行するユースケースを実装して評価を行った（**図-7**）。人の数は最大40万人，各人が5秒に1回位置情報をあげるモデルとした。このモデルにおいて，**図-8**に示すように，各人の位置・嗜好情報を抽出し（クエリ1），各人と一定距離以内の店舗情報を抽出し（クエリ2），各人の嗜好情報と抽出店舗の商品情報をマッチングさせ（クエリ3），マッチングした場合にクーポンを発行する。人の数（入力イベント）の増加に伴い，処理負荷が大きく増大するクエリ2・3に対して，スムーズに途切れなくサーバを追加し（例えば，4台の処理サーバを8台に増加），CPU使用率を低減

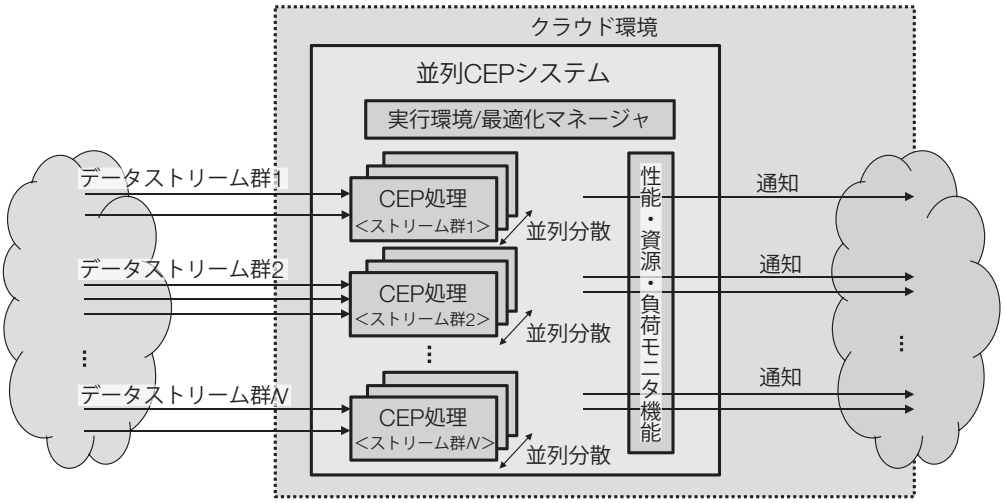


図-4　並列CEPシステムの構成
Fig.4-Configuration of parallel CEP system.

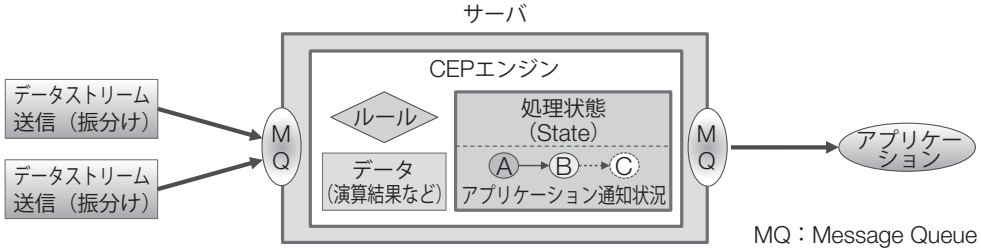


図-5　CEPの基本構造
Fig.5-Basic structure of CEP.

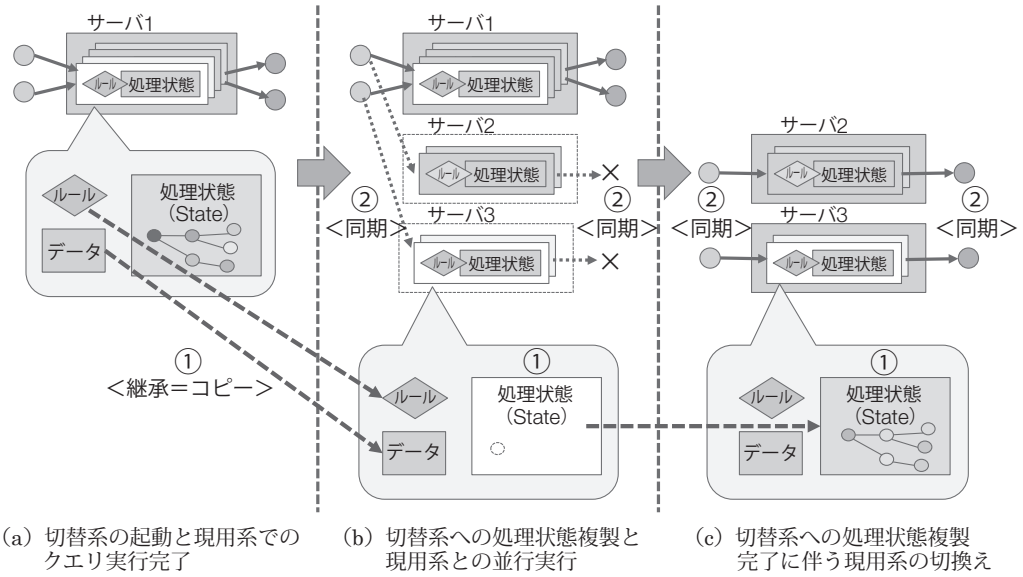


図-6　CEP動的負荷分散方式(現用系から切替系への移行)
Fig.6-CEP dynamic load balancing method (transition from current system to extra system).

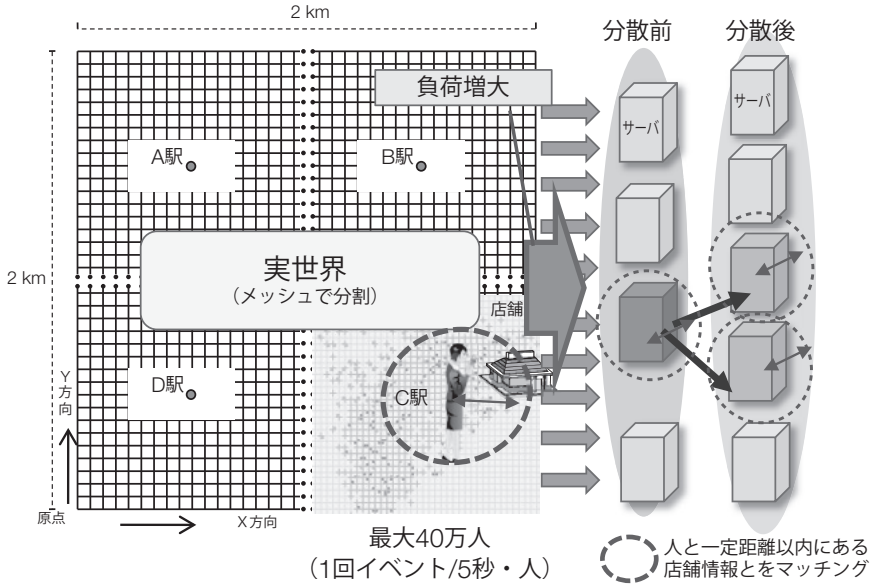


図-7　CEP動的負荷分散の評価システム
Fig.7-Evaluation system of CEP dynamic load balancing.

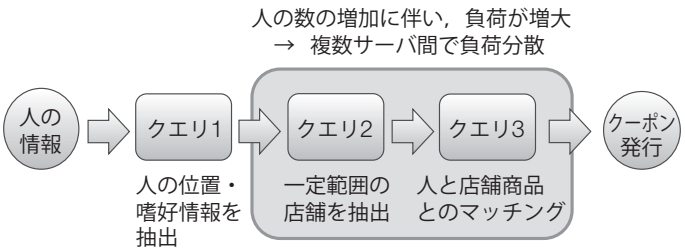


図-8　評価システムにおける使用ルール(クエリ)
Fig.8-Actual rules (query) in evaluation system.

し，高速な処理性能を保証できることを確認している。

本実装により，GC（Garbage Collection）による処理のゆらぎやイベント配信同期など，いくつかの課題も明確になっている。今後は，これらの課題を解決しながら，CEP動的負荷分散機能における性能や資源効率の向上，および運用機能との連携による最適化などに取り組んでいく。

データ処理ワークフロー記述言語

「価値ある情報」を見出すには，性能や容量のみならず，原データに対して抽出や集計などの様々な処理を繰り返し試行することが必要で，簡易で生産性が高い開発環境が必要である。

DISPEL（Data-Intensive Systems Process Engineering Language）は，高レベルのデータ処理ワークフローを記述する言語である。この言語は，欧州富士通研究所が参加しているADMIREプロジェクト（EU Framework 7）で開発され，データ指向コンピューティングのアーキテクチャおよびプラットフォームによって大量データの利用を促進し，ユースケースを通じて知識発見を支援することを目的としている。

● 言語の概観

DISPELはJavaに似た仕様により，ストリーミング指向のデータワークフローを記述する分散システム向け言語である。

一例として，地震データベースの処理を挙げる。ここでは，過去24時間の間に蓄積された原データを変換し，別のデータベースに格納している。DISPELでは，この変換作業を抽象化したコンポーネントを使い，つぎのようにシンプルに表すことができる。

```
use dispel.db.SQLQuery;
use dispel.lang.Results;
use eu.seismo.Transform;

SQLQuery sq  = new SQLQuery;
Transform tr = new Transform;
Results res  = new Results;

sq.data   => tr.input;
tr.output => res.input;
|-"uk.ac.bgs.earthquakes"-| => sq.source;
|-"SELECT FROM WHERE"-| => sq.expression;
|-"Last 24 hours"-| => res.name;

submit res;
```

本言語のキーとなる機能は以下のとおりである。

(1) 型システムを3レベル（開発者およびデータエンジニア向け，地震学者などのエンドユーザ向け）持つことにより，関心の分離を図っている。
(2) 強力な高階関数により，処理エレメント間の複雑なワークフローを表現している。
(3) データフローのアノテーションにより，情報フロー，プロセス終了およびエラー管理をサポートしている。

これらの機能により，地震学者と具体処理の開発者といった抽象度の異なる対象を扱う人々の間での共同作業がスムーズで誤りが少なく実行できるようになる。

● 欧州でのDISPELユースケース

(1) Platform for Analytical Customer Relationship Management（ACRM）

このユースケースでは，DISPELをテレコムデータに適用し解約予測を行う。目的は，電話の利用者がほかの会社に移行する理由を発見し，移行しないように適切な対抗手段を準備するためである。このACRMの開発では，どのようなデータセットにするかと，モデルをどのように準備し評価するかが課題であった。データは4種類の異なるデータベースから構成される。

・Customer DB：年齢，性別
・Contract DB：契約数，料金プランの推移
・Contact DB：コールセンタで受けたクレームや質問の数
・Call DB：夕方に発話した数

多くの場合，ワークフローで重要なのは，複雑なデータマイニングを行う前に，データを準備し利用可能な形に修整すること（クレンジング）である。

通常の業務では，ユーザはDISPELコードとして記述されるデータ指向エンジニアリングプロセスの詳細ではなく，ポータルを通じて操作を行う。例えば，データ指向エンジニアはDISPELを使っ

て業務ドメインのエキスパートに複数のモデルからの選択肢を与え，どの戦略が業務レベルで最善かを選ばせることができる。さらに，ドメイン間で多くのエンジニアリングパターンが共通のため，DISPELにおける知識発見プロセスは再利用を促進することができる。

ADMIREプロジェクトでは，一つのDISPELコードがすべてのワークフローの詳細，すなわちデータ取得，変換，クレンジング，フェデレーション，マイニングと処理結果の取得を表現するため，複雑なワークフロー管理を簡単化することができた。

（2）Data-Intensive Seismology

地震波のデータが増えて利用可能になるにつれ，プロセスよりもデータを中心に考えるデータ指向研究の機会が増えてきた。

IRIS（Incorporated Research Institutions）におけるDMC（Data Management Centre）アーカイブでは，いくつもの国際的な地震学ネットワークからのデータを収集し，その量は年間21 Tバイトに至っている。これは，観測所で計測された地震波からスタンドアローンで分析するという伝統的な手法が困難になってきていることを意味する。ここではDISPELを使い，データ取得，前処理，相関処理（cross-correlate）を行い，そして何Tバイトものデータを二つの地震データアーカイブ（British Geological SurveyおよびORFEUS Data Centre of the European Seismological Commission）に保管している。

技術上，表面波散布計測の周囲騒音（ambient noise）データ分析は次の4ステップで実装される。

（1）単一ステーションデータ準備
（2）相関および一時的データ保持
（3）散布カーブの計測
（4）品質コントロール

DISPELワークフローは全体プロセスをコントロールする。複数のデータソースからの情報を同時に分析することで，これまでよりも地殻や地震に対する科学的な知見が得られやすくなってきている。

むすび

クラウドは，業務プロセスのICT化から，クラウドに集約されるデータの分析処理によって「価値ある情報」を見出して，売上拡大や社会システム最適化を行う「イノベーション」へと拡大しようとしている。

イノベーションがこれまでのICT化と大きく異なる点の一つは，イノベーションではやるべきことが分からないところから出発し，集まった大量データを様々に分析しながら徐々にやるべきことが分かっていく，ということである。そのため，データ分析は違った観点から何度も繰返しが必要であり，開発と運用（故障対応から蓄積コストまで）のあらゆる局面において高速，かつ，低コストな処理が必要となる。クラウドが提供する，一時的な大量の計算資源の利用，リソース共用による低コスト化，といった利点は，まさにこの要請に応える可能性を持っており，すでに提供が始まっている富士通のクラウドサービス上に，大量データ処理向けの高速で低コストな処理，簡易で高生産性のデータ活用向け開発環境を実現すれば，イノベーションの新しい市場開拓につながるはずである。

富士通および富士通研究所では，大量データ処理の基盤技術である分散データストアや複合イベント処理の並列化に取り組んでおり，分散環境での高速な集計やサービスを止めずに増設する技術を開発した。また，欧州富士通研究所では大量データの分散処理のための記述言語の研究を行っており，データ分析作業をスムーズにして科学研究の発展に貢献している。今後は，これらの要素技術や実践を充実・発展させながら，クラウド環境で動作する様々な機能を密接に連携させて，複合的な処理を迅速に開発・実行するための技術を提供していく予定である。

なお，複合イベント処理（CEP）の並列化については，経済産業省殿の平成22年度産業技術研究開発委託費「次世代高信頼・省エネ型IT基盤技術開発事業」における委託業務の成果を利用している。

参考文献

（1）Big data to drive a surveillance society - Computerworld.
http://www.computerworld.com/s/article/9215033/Big_data_to_drive_a_surveillance_society

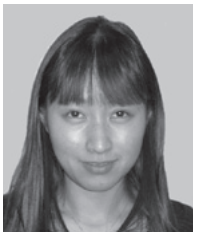
(2) 富士通：クラウドサービス向け分散型データ保管技術の高速化に成功.
http://pr.fujitsu.com/jp/news/2010/06/17.html

著者紹介

土屋　哲（つちや　さとし）
クラウドコンピューティング研究センター 所属
現在，並列分散処理技術全般の研究開発に従事。

槌本裕一（つちもと　ゆういち）
クラウドコンピューティング研究センター 所属
現在，並列分散処理技術，分散データストアの研究開発に従事。

坂本喜則（さかもと　よしのり）
クラウドコンピューティング研究センター 兼 クラウドプラットフォーム開発本部コンバージェンスサービスPF開発統括部 所属
現在，並列分散処理技術，複合イベント処理の研究開発に従事。



Vivian Lee
欧州富士通研究所 所属
現在，大量データ処理の開発環境の研究開発に従事。