

# スケーラブルな複合イベント処理技術

## Scalable Complex Event Processing Technologies

磯山 和彦  
Kazuhiko Isoyama

喜田 弘司  
Koji Kida

NEC クラウドシステム研究所  
Cloud System Research Laboratories, NEC Corporation

This paper introduces the concept of Complex Event Processing (CEP) and describes its scalability issues when it is applied to Big Data environments. The paper also introduces our scalable CEP technologies which realize high throughput and large scale CEP.

### 1. はじめに

近年、電力消費量や交通情報など、様々な情報を収集するセンサや、ネットワークカメラやスマート家電などのようにネットワークを経由して遠隔操作を行えるデバイス(アクチュエータ)がネットワークにつながれるようになってきている。それに伴い、センサからの情報を基に、人手を介することなくデバイスを制御する“M2M (Machine to Machine)”の技術が注目を集めている。

今後、このようなネットワークに接続されるデバイスは急増し、そこから発生する情報は膨大な量になると予想される。このような膨大な情報は総じて“ビッグデータ”と呼ばれる。

ビッグデータは、そのままでは単純な意味しか持たない大量のデータに過ぎないが、その膨大な情報の中から価値のある情報を発見し、組み合わせることによって、意味のある有用な情報として活用することができる。このような技術の一つに“複合イベント処理(CEP:Complex Event Processing [Luckham 02])”がある。

本稿では CEP について解説するとともに、筆者らが提案するビッグデータに対応するスケーラブルな CEP 技術について紹介する。

### 2. CEP

#### 2.1 イベントとは

CEP の分野では、情報、情報の発生、状態の変化または出来事のことを“イベント”として統合的に扱う。イベントはその概念レベルにより下位～上位に分類される。例えば下位のイベントとしてはセンサ値の読み出しイベント、ビジネス上の 1 トランザクションなどが挙げられる。また、上位のイベントとしては人の状態(健康、気分)、ビジネスの状態(取引完了、契約違反)などが挙げられる。

#### 2.2 CEP とは

人間の活動のような上位のイベントは、下位のイベントの組み合わせから推測することができる。このように一つひとつの単純なイベントからは推測できないイベントを、複数のイベントの組み合わせから推測し、処理を行う技術が CEP である。

CEP ではこのように検出したいイベントの組み合わせをイベントパターンとして記述し、そのイベントの組み合わせを発見したときに行いたい処理をアクションとして、“IF イベントパターン THEN アクション”の形式で記述し(CEP ルール)、それに基づいて処理を行う。

### 3. ビッグデータにおける CEP の課題

ビッグデータは、1 台のサーバ(EP: Event Processor)で CEP を行うにはデータ量が膨大であるという問題がある。それに対処するためには CEP の処理の負荷を分散することが必要となる。

サーバ処理を分散させるためには、従来からロードバランサが使用されていた(図 1-a)。ロードバランサでは入力データを分配する方法としてはラウンドロビンなどを使用することができるが、CEP は過去のイベントを“ステート”に保持して新たに受信したイベントと組み合わせる処理を行う“ステートフル”な処理であるため、このようなラウンドロビンによる負荷分散はそのままでは適用できないという課題がある。例えば図 1-a では A と B のイベントが別々の EP に送られるため、CEP ルール“A&B”を発見することができない。

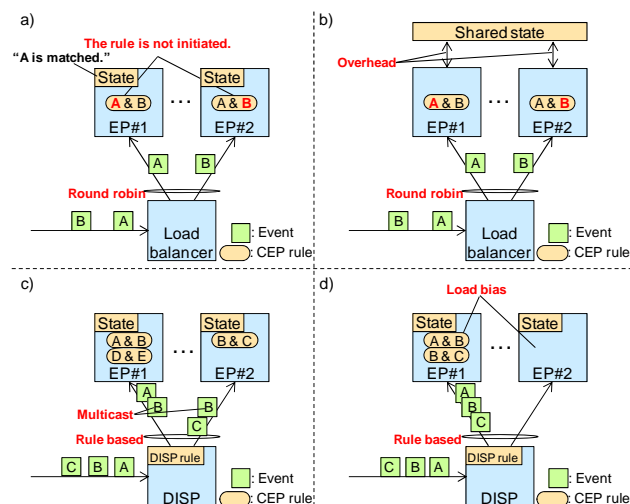


図 1 CEP 負荷分散方式のバリエーション

これを解決するためには全ての EP が情報(ステート)を共有する必要がある(図 1-b)。図 1-b では各 EP は CEP ルールの部分マッチ情報を共有ステートに保持し、他の EP がその情報にアクセスすることができるようにすることにより、イベント A と B が別の EP に転送されたとしても、CEP ルール“A&B”が成立したことを発見することができる。

しかしながら、この方式では、共有ステートは EP の外部のメモリに存在するため、アクセスにオーバーヘッド(遅延、スループット)があるという問題がある。この問題を解決するためにはステートを EP 内に局所化する必要がある。

図 1-c はそれを解決する、我々の提案するシステムが採用する方式である。このシステムでは従来のロードバランサの代わりにイベントディスパッチャ(DISP)を採用する。DISP はイベントディスパッチルール(DISP ルール)に基づいてイベントを転送する。各 DISP ルールにはイベント条件と宛先 EP のペアが記述されている。システムは各 CEP ルールを一つの EP に配置し、そのルールに関するイベントはその EP に転送されるように DISP ルールを設定する。このようにすることにより、ステートを EP 内に局所化することが可能となる。

しかし、もし CEP ルールが何も考慮されずに分配されると、図 1-c のイベント B のように DISP においてマルチキャストが多発し、システムスループットを低下させる原因になるという別の問題が生じる。これを避けるためには、同じイベントに関する CEP ルールはできる限り同じ EP に分配されることが望ましい。

一方で、もし同じイベントに関する全ての CEP ルールが一つの EP に分配されると、EP 間で負荷の不均衡が生じるという課題がある。図 1-d では CEP ルール“A&B”と“B&C”は EP#1 に配置され、イベント B のマルチキャストは回避されるが、EP#1 が高負荷となり、EP#2 が低負荷となってしまっている。

よって、EP 間で負荷をバランスさせつつ、DISP におけるマルチキャストを最小化させるという、相反する課題を解決する CEP ルール分配アルゴリズムが必要となる。我々は[Isoyama12]においてこのような CEP ルール分配アルゴリズムを提案した。

#### 4. ルール分配アルゴリズム

前節の課題を解決するために、我々のルール分配アルゴリズムは、1)同じイベントを必要とする EP 数を最小にする、という制約の下、2) EP 間の処理負荷の差をある一定の範囲内にする。

1)によって DISP におけるマルチキャストを最小にし、同じイベントに関するステータを持つ EP を最小にすることができ、2)により、処理負荷がある EP に偏ることを避けることができる。

図 2 にルール分配アルゴリズムの動作概要を示す。

ルール分配アルゴリズムでは、図 2 のようにルールが扱うイベント条件の間に依存関係のあるルールをグルーピングし、グループ内のルールが分離しないように、サーバにルールを分配する。このとき、ルール 3 のように複数のルールと別個に依存関係がある(ルール 3 はイベント A&B を扱い、ルール 1 はイベント A&C を、ルール 8 はイベント B&D を扱うなど)場合、なるべくそれらのルールは同じサーバに分配する。

しかし、依存関係のグループが数珠つなぎになると、グループが大きくなりすぎ、そのグループのルールを一つのサーバに配置すると、そのサーバの負荷だけが大きくなってしまいう問題がある。その場合はグループを分割し、別のサーバに配置する。分割により両方のサーバに転送する必要のあるイベントは DISP がマルチキャストする。

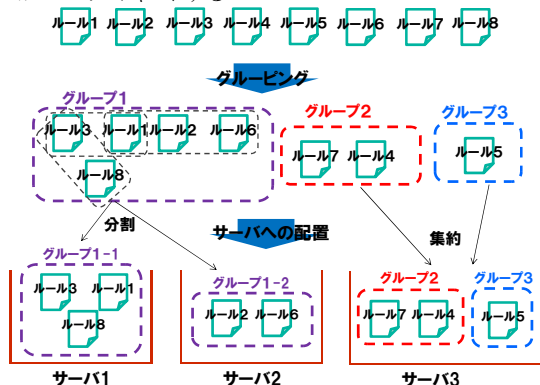


図 2 ルール分配アルゴリズム概要

一方でルール 5 とルール 4,7 のグループのように依存関係のないルールを同じサーバに分配することにより各サーバの負荷のバランスを整える。

#### 5. スケーラビリティ評価

最後に我々のスケーラブル CEP システムの特性を評価した。

##### 5.1 評価環境

評価では DISP として 4 台のサーバを用意した。それぞれのサーバは 4 つの Intel® Xeon X7460 (2.66 GHz, 6 コア)と 128 GB RAM を装備する。また、EP として 6 台のサーバを用意した。それぞれのサーバは 2 つの Intel® Xeon X5355 (2.66 GHz, 4 コア)と 16 GB RAM を装備する。

##### 5.2 パラメータ

パラメータとしては EP のプロセス数を 4~48 と変化させ、最大スループットを計測した。総 CEP ルール数は N=10,000 とした。また、パラメータとして、入力イベントが CEP ルールにマッチするレート  $m$  を設定した。評価では入力イベントの属性値の取り得る範囲を調整することによって  $m=0.1$  または  $0.5$  に変化させて計測した。 $m$  が大きくなると DISP でフィルタアウトされない(=DISP を通過して EP での処理対象となる)イベントが多くなり、同じイベント量でもシステムにかかる負荷がより大きくなる。

##### 5.3 評価結果

図 3 に評価結果を示す。図 3 によると、EP のプロセス数に比例してシステムのスループットが線形に増加していることが分かる。EP が 48 プロセスのとき、 $m=0.1$  の場合 270 万イベント/秒、 $m=0.5$  の場合 220 万イベント/秒のスループットを計測した。

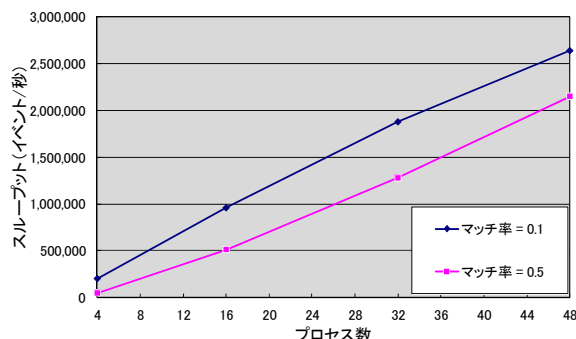


図 3 スループット - EP プロセス数。

#### 6. むすび

本稿では CEP について解説するとともに、ビッグデータに CEP を適用する際の課題を整理し、筆者らが提案するビッグデータに対応するスケーラブルな CEP 技術について紹介した。

このような技術を採用することにより、今後、本格化し、大規模化していく M2M サービスに対応できるイベント処理を提供できると考えている。

#### 参考文献

[Luckham 02] Luckham, D.: The Power of Events, ISBN 0-201-72789 (2002)  
 [Isoyama 12] Isoyama, K., Kobayashi, Y., Sato, T., Tagato, H., Kida, K. and Yoshida, M.: A scalable complex event processing system and evaluations of its performance, 6th ACM Int. Conf. on Distributed Event-Based Systems (DEBS2012) (2012)