

## 時空間イベントを扱う複合イベント処理システム

高橋 正和<sup>†</sup> 稲葉 鉄平<sup>§</sup> 石川 佳治<sup>††</sup>

<sup>†</sup> 名古屋大学工学部電気電子・情報工学科 <sup>§</sup> <sup>††</sup> 名古屋大学大学院情報科学研究科

<sup>††</sup> 国立情報学研究所

### 1 はじめに

近年、ビッグデータと呼ばれる巨大なデータを高速に処理することが必要となってきている。それに関連する要素技術の1つとして、**複合イベント処理** (complex event processing, CEP) がある [1, 2, 3]。CEP はデータストリームから検出されるイベントを複合的に組み合わせ高次のイベントを処理するための技術で、現在大いに着目されている。一方、今日では GPS などのさまざまなセンサから実世界のデータを大量に取り込むことができる。

本研究では、特に時間情報・空間情報が伴う**時空間イベントストリーム**に着目し、それらに対する複合イベント処理を考える。時空間データに対するフィルタリング条件の適用などを含む複合イベント処理を記述するためのシステムを開発する。本システムは、マイクロソフトの複合イベント処理フレームワークである *StreamInsight* [5] 上に構築する。時空間イベントを扱うための機能を追加し、LINQ [4] 形式による複合的な時空間イベント処理を可能とする。

### 2 システムの概要

#### 2.1 基本的なアイデア

データストリームから抽出されたイベントの処理は、研究および実応用でも近年大いに注目されている。本研究では、イベント処理の幅広い応用分野の中から、時空間イベントに着目することにした。自動車におけるナビゲーションや、携帯機器を持ったモバイルユーザに対する情報提示など、さまざまな応用を考えることができる。以下の点がポイントとなる。

- プログラミング言語との親和性が高いこと：イベント処理システムを構築する場合、データベースの間合せ言語のような独自の高レベルの言語を設計することが一つのアプローチであるが、本研究ではプログラミング言語に埋め込む形で密に連携するアプローチを採用する。イベント処理では、ユーザ独自にフィルタリング条件を追加したりすることなどがあるため、利便性がより高いと考えられるためである。

- 空間イベントを扱えること：空間的な領域が対応づけられたイベントのことを本研究では**空間イベント** (spatial event) と呼ぶ。実際には、イベントには時間的情報も含まれることから、そのことを強調する場合には**時空間イベント** (spatio-temporal event) とも呼ぶ。空間イベントに対しては、フィルタリング処理やイベントの合成に関するメソッドを対応づけ、時間的、空間的、またそれらを組みあわせたセマンティックに基づく複合イベント処理を記述できるものとする。

前者の考察に基づき、本システムではマイクロソフト社の *StreamInsight* [5] を基盤にシステムを開発することにした。*StreamInsight* の基本概念については以下で述べる。後者については、*StreamInsight* 上での時空間イベントに関する複合イベント処理を記述を可能とするクラス群を構築することにした。イベント処理の中心となる部分については、LINQ [4] 形式による分かりやすい表現が可能となる。

#### 2.2 StreamInsight について

マイクロソフト *StreamInsight* は、SQL Server 2008 R2以降に含まれるソフトウェアパッケージであり、複合イベント処理を含むアプリケーションを容易に記述するための機能が提供されている。イベント列は Input アダプタにより取得され、CEP Server 上の待ち行列に入れられる。入力イベント列を処理するために LINQ クエリを使用する。加工した結果は Output アダプタにより出力される。

*StreamInsight* を用いた場合、C#などのプログラミング言語に LINQ クエリ形式の複合イベント処理を自然に埋め込むことができるという利点を持つ。そこで、本研究ではこれをベースにシステムの開発を行う。

#### 2.3 システムの構成

システムの構成を図1に示す。時空間 CEP アプリケーションはC#プログラムとして記述される。プログラム記述においては、本システムが独自に提供する時空間イベント関係のクラスが利用できる。複合イベント処理は LINQ クエリの形式に似た形で記述される。ただし、*StreamInsight* に構文上の制約があることから、時空間 CEP 処理記述はプリプロセッサを用いて、*StreamInsight* が処理できる形の CEP プログラムに展開する。CEP プログラム中でのイベント処理に相当する LINQ クエリの部分が *StreamInsight* が

Complex Event Processing System for Spatio-temporal Events

Masakazu Takahashi<sup>†</sup>, Tepei Inaba<sup>§</sup>, Yoshiharu Ishikawa<sup>††</sup>

<sup>†</sup> Department of Information Engineering, School of Engineering, Nagoya University

<sup>§</sup> Graduate School of Information Science, Nagoya University

<sup>††</sup> National Institute of Informatics

提供する CEP サーバにおける連続的問合せとして登録され実行される。

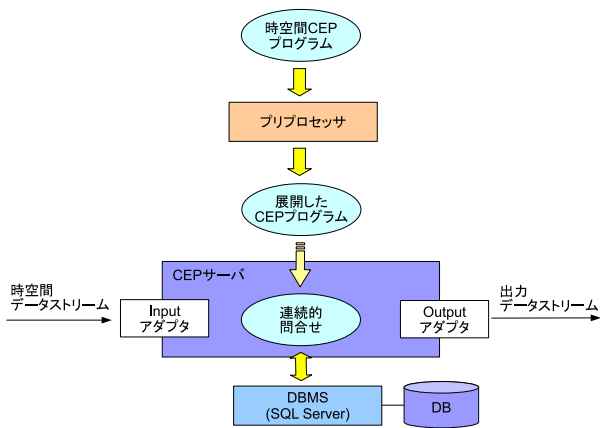


図 1: システムの構成

本システムでは、時空間イベントとデータベースに蓄積された空間データとの連携も行う。マイクロソフト SQL Server には空間データ管理のための機能が含まれているが、この機能を活用することで、発生した時空間イベントに応じて、時間的・空間的なセマンティクスに基づき、データベース中の空間データとの統合なども可能とする。

### 3 複合イベントの処理手法

#### 3.1 時空間イベント処理の記述例

時空間イベント処理を記述する例 (C#プログラムの一部) を図 2 に示す。まず、入力ストリームを生成している。EventPayload クラスは、イベントを保持する汎用クラスである。CepStream クラスはイベントストリームに関するものであり、ここでは Create メソッドによりストリームを生成している。

その下の個所では、矩形領域を表す areaA を設定し、入力ストリームから、areaA に出現位置が含まれるようなイベントのみを抽出し、それを出力ストリーム outputStream に出力している。from から select の部分は LINQ の構文にしたがったフィルタリング記述となっている。メソッド InRect は、そのイベントが指定された矩形領域に含まれるかを判定する。

図 2 のイベント処理記述は、StreamInsight システム上ではそのまま実行することはできない。LINQ 形式の中では、ユーザ定義のクラスやそれに伴うメソッドを自由に記述できるわけではなく、さまざまな制約が課される。これはプログラムの安全性や実行可能性のためであるが、時空間イベントに関する処理を記述することは難しくなる。そのため、本研究では図 2 のような記述をプリプロセッサにより展開し、StreamInsight が処理できるプリミティブなクラスに展開する。また、LINQ クエリ構文をより柔軟であるメソッド構文に書き換える。

```
// 入力ストリームの生成
var inputStream
  = CepStream<EventPayload>.Create(...);
...
// 矩形領域によるフィルタリング
Rect areaA = new Rect(20, 100, 20, 60);
var outputStream
  = from e in inputStream
    where e.InRect(areaA)
    select e;
```

図 2: イベント処理の例

#### 3.2 データベースとの連携

また、本システムではデータベースとの連携も図る。図 3 は、イベントの発生場所を含むような地域情報を付与する例を示す。StreamInsight 自体、LINQ の機能を用いて SQL Server と連携することは基本的には可能である。ただし、SQL Server が有している空間型データの処理機能を直接的には利用できない。そこで本研究では、図 3 の included のようなメソッドをプリプロセッサで SQL Server が提供する空間型データメソッドに展開して呼び出す。

```
// データベースのオープン
DataContext db = new DataContext(...);
Table<Area> areas = db.GetTable<Area>();
...
// データベースとの結合処理
var outputStream
  = from e in inputStream
    from a in areas
    where e.included(a)
    select new {e, a};
```

図 3: データベースとの連携

### 4 まとめ

本稿では、時空間イベント処理システムの概要について述べた。今後はその詳細化とシステム実装について取り組みたい。

#### 謝辞

本研究は内閣府最先端研究開発プロジェクト (FIRST) および科研費 (22300034) の助成による。

#### 参考文献

- [1] G. Cugola and A. Margara. Processing flows of information: From data stream to complex event processing. *ACM Comput. Surv.*, 44(3), 2012.
- [2] O. Etzion and P. Niblett. *Event Processing in Action*. Manning, 2011.
- [3] 磯山, 佐藤, 喜田. スケーラブルな複合イベント処理技術. *人工知能学会誌*, 28(1):91-97, 2013.
- [4] LINQ: .NET 統合クエリ. [msdn.microsoft.com/ja-jp/library/bb308959.aspx](http://msdn.microsoft.com/ja-jp/library/bb308959.aspx).
- [5] Microsoft StreamInsight. [msdn.microsoft.com/ja-jp/library/ee362541.aspx](http://msdn.microsoft.com/ja-jp/library/ee362541.aspx).