

ECA ルールを用いた配信型情報源統合環境の構築

7W-03

渡辺 陽介† 梶野 智行†† 北川 博之††† 石川 佳治†††

† 筑波大学 第三学群 情報学類 †† 筑波大学 理工学研究科 ††† 筑波大学 電子・情報工学系

1 はじめに

近年の情報技術の発展によって我々の身近に配信型情報源が出現しており、それに伴って配信型情報源を統合利用する必要性が高まっている。例えば、Web 上で企業ニュース配信サービスと株式情報配信サービスが提供されており、ユーザは自分の所有する株式の情報をリレーショナルデータベースに格納しているとする。このような場合、ユーザが所有している IT 関連企業の株価が指定した閾値以上になったら、その企業名と株価、企業に関連したニュース記事 2 日分をつけて翌日の午前 0 時に配信して欲しいといった要求が考えられる。

配信型情報源では配信サーバから情報が随時到着するため、情報の到着や時間の経過などのイベントの発生に応じて自動的に情報の蓄積や廃棄などの処理を行なう必要がある。我々の研究グループが構築した配信型情報源統合環境 [1][2] では、発生する各イベントに対応した処理を ECA ルール [3] として記述して与えることで要求が実現できるようになっている。しかし、要求の実現には互いに連係して動くような ECA ルールを複数記述する必要がある。ユーザがそのような作業を行うことは困難であるので、より宣言的な記述から ECA ルールを自動生成することが望ましいと考えられる。本稿では、ユーザが記述したリレーショナル代数式から ECA ルールを生成する枠組の実現について述べる [4]。

2 統合システム概要

2.1 アーキテクチャ

本研究では、アーキテクチャとしてメディエータ/ラッパー方式を採用している (図 1)。情報源ごとに対応するラッパーがあり、ラッパーは情報源に対する問合せなどの処理を請け負う。ラッパーの上位にはメディエータがあり、情報の統合処理を行なう。配信型情報源ラッパーは、情報配信サーバからの情報 (配信単位) が送られてくると、情報が到着したことを知らせるための arrival イベントを発生する機能を持っている。他にイベントを発生させるモジュールとしてタイマーモジュールがあり、指定した時間が来たことを知らせる alarm イベントを発生させる。ルール処理モジュールでは統合・配信処理を記述した ECA ルールを保持している。イベントが発生すると、各イベントに対応する ECA ルールを評価・処理し、メディエータに統合処理要求を送ったり、配信モジュールに統合結果の配信要求を出したりする。本稿で取り上げるルール生成モジュールは、ユーザに与えられた配信要求記述から処理に必要な ECA ルールを生成してルール処理モジュールに与える役割を持つ。

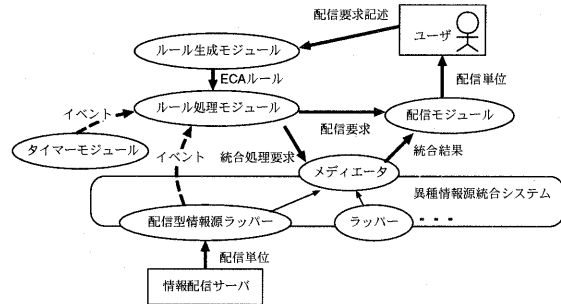


図 1: システムアーキテクチャ

2.2 データモデル

本システムは各情報源をリレーションとしてモデル化しており、配信型情報源も同様にリレーションとして扱う。配信サーバから到着した配信単位はそのリレーションに属する 1 タプルとして順にシステムの一時領域に格納される。我々は統合の対象となるこれらのリレーションを I-sequence relation と呼んでいる。統合結果もリレーショナルモデルで表され、統合結果の各タプルは配信予定時刻が来るとユーザに配信される。統合結果となるリレーションを O-sequence relation と呼ぶ。I-sequence relation は、その情報源に固有な属性の他に、タプルが到着した時刻を表すための ITS 属性を持つ。同様に O-sequence relation はタプルの配信予定時刻を表す OTS 属性を持つ。ITS・OTS 属性とともに時刻ドメインの値である。

2.3 ECA ルール

ECA ルールはアクティブデータベースで使われているものと同じで、あるイベントに応じた処理を記述するのに用いられる。一つのルールはイベント節 (on 節)、コンディション節 (if 節)、アクション節 (do 節) からなる。配信型情報源の統合には以下の 3 段階の処理が必要である。

配信情報の蓄積: 到着した配信単位がユーザの要求を満たすのに必要なら、I-sequence relation に追加する。この処理を行なうルールを **storage rule** と呼ぶ。

新たな配信情報の生成: 情報を統合して要求された配信単位を生成する。この処理を行なうルールを **generation rule** と呼ぶ。

不要な配信単位の廃棄: I-sequence relation から将来使用されないことが明らかな配信単位を定期的に削除する。この処理を行なうルールを **garbage disposal rule** と呼ぶ。

3 ECA ルールの自動生成

3.1 配信要求記述

ユーザは配信要求要求を以下のような代数式で記述する。

$$O_{new} = \Omega_{f(I_k, ITS)}(E(I_1, \dots, I_n))$$

O_{new} は新しく定義される O-sequence relation を表す。E は統合操作を指定するリレーショナル代数式で、 I_1, \dots, I_n

Development of a Dissemination-Based Information Integration Environment with ECA Rules

Yousuke Watanabe†, Tomoyuki Kajino††

Hiroyuki Kitagawa†††, Yoshiharu Ishikawa†††

† College of Info. Sci., Third Cluster of Colleges, Univ. of Tsukuba

†† Master's Degree Program in Sci. and Eng., Univ. of Tsukuba

††† Institute of Info. Sci. and Elec., Univ. of Tsukuba

は統合対象となる I-sequence relation である。Ω は統合結果から ITS 属性を除去して OTS 属性を付加する演算である。f は配信単位の到着時刻から配信予定時刻を求める関数で、f(I_k.ITS) は配信予定時刻を表す。この関数の引数になった、配信予定時刻の決定に関与する I-sequence relation I_k をマスタ情報源と呼ぶ。f には以下の関数の組合せを記述する。

next_p(t): t に最も近い未来に条件 p を満たす時刻
previous_p(t): t に最も近い過去に条件 p を満たす時刻
after_{δt}(t): t から時間 δt 経過した時刻
before_{δt}(t): t から時間 δt 遡った時刻

3.2 処理可能性

例えば「翌日届くニュースを今日配信せよ」という要求を記述されてもそれは実行不可能である。よってユーザの要求が処理可能なものを調べる必要がある。配信要求 O_{new} = Ω_{f(I_k.ITS)}(E(I₁, ..., I_n)) が与えられたとき、次の条件を満たすものは **consistent** であるという。

1. 代数式 σ_{I_k.ITS ∈ f⁻¹(t)}(E(I₁, ..., I_n)) の選択演算を push することで E((σ_{I₁.ITS ∈ Ψ₁(t)}(I₁)), ..., (σ_{I_n.ITS ∈ Ψ_n(t)}(I_n))) という形に変形可能である。ただし Ψ_i は各 I₁, ..., I_n ごとの時刻についての選択条件である。
2. 時刻に関する選択条件 Ψ_i の中には配信予定時刻より未来を指し得る条件が存在しない。

f⁻¹(t) は配信予定時刻を求める関数 f(t) の逆関数を表す。

$$f^{-1}(t) = \{u \mid f(u) = t\}$$

各関数に対応する逆関数は以下の通りである。

next_p⁻¹(t) = {u | previous_p(t) ≤ u < t}
previous_p⁻¹(t) = {u | t ≤ u < next_p(t)}
after_{δt}⁻¹(t) = {u | u = before_{δt}(t)}
before_{δt}⁻¹(t) = {u | u = after_{δt}(t)}

3.3 廃棄可能性

例えば「ある企業に関する過去すべてのニュース記事をその企業の創立記念日に配信せよ」という場合、蓄積された情報は将来に渡って廃棄する必要がない。まず、ある時刻 t 以降に必要となる配信単位の集合を以下のように表現する。

$$\text{TSet}(\Psi_i, t) = \bigcup_{\tau > 0} \Psi_i(t + \tau)$$

$$\text{TSet}^+(\Psi_i, t) = \bigcup_{\tau \geq 0} \Psi_i(t + \tau)$$

このとき、配信要求記述が廃棄可能なデータを含み得るかを、以下の条件で判定する。

$$\Psi_i(t) - \text{TSet}(\Psi_i, t) \neq \emptyset$$

3.4 処理手順

入力として配信要求記述 O_{new} = Ω_{f(I_k.ITS)}(E(I₁, ..., I_n)) が与えられる。

1. 配信要求記述を内部形式に変換する。
2. 配信要求記述を σ_{I_k.ITS ∈ f⁻¹(t)}(E(I₁, ..., I_n)) に変換する。
3. 選択演算を Push して以下の形式に変換する。C_i は I_i の選択条件のうち、時刻以外についての条件を表す。
E'((σ_{C₁}(σ_{I₁.ITS ∈ Ψ₁(t)}(I₁))), ..., (σ_{C_n}(σ_{I_n.ITS ∈ Ψ_n(t)}(I_n))))
4. 配信要求が consistent であるかを調べ、consistent であるなら以下の処理に移る。そうでなければ処理不可能として終了する。

5. storage rule, generation rule, garbage disposal rule の生成。garbage disposal rule については廃棄可能なデータを含んでいる場合のみ生成する。

3.5 ECA ルールの生成規則

storage rule: storage rule は I-sequence relation ごとに生成される。now は現在時刻、Temp_{I_i} は I_i の一時領域を表している。I_i がマスタ情報源の場合には、配信予定時刻に alarm イベントを発生させる必要があるので setTimer 演算でタイマーをセットする。

(i) I_i がマスタ情報源の場合

Rule Storage_i

on: arrival(I_i)
if: I_i.ITS ∈ TSet⁺(Ψ_i, now) ∧ C_i
do: Temp_{I_i} + = I_i ;
 setTimer(f(I_i.ITS), New) ;

(ii) それ以外の場合

Rule Storage_i

on: arrival(I_i)
if: I_i.ITS ∈ TSet⁺(Ψ_i, now) ∧ C_i
do: Temp_{I_i} + = I_i ;

generation rule: generation rule は 1 つの配信要求記述につき 1 つ生成される。Deliver は統合結果の配信を行なう演算である。

Rule Generation_i

on: alarm(New)
if: true
do: O_{new} = E'(Temp_{I₁}, ..., Temp_{I_n});
 Deliver(O_{new});

garbage disposal rule: 廃棄ルールは、廃棄可能なデータを含み得る I-sequence relation のみそれぞれ生成される。INT は廃棄ルールを呼び出す間隔を表す。

Rule GarbageDisposal_i

on: alarm(GarbageDisposal_i)
if: true
do: Temp₋ = σ_{Temp_{I_i}.ITS ∈ TSet⁺(Ψ_i, now - INT)}
 -TSet⁺(Ψ_i, now)(Temp_{I_i});
 setTimer(now + INT, GarbageDisposal_i);

4 おわりに

本稿では配信型情報源統合環境において新たな配信サービスを定義するための枠組と、ユーザが与えた配信要求記述から必要な ECA ルールを生成するための枠組について述べた。

参考文献

- [1] H. Kitagawa, A. Morishima and H. Mizuguchi. Integration of Heterogeneous Information Sources in InfoWeaver. *Advances in Database and Multimedia for the New Century - A Swiss / Japanese Perspective*, World Scientific Publishing, pp. 124-137, 2000.
- [2] H. Mizuguchi, H. Kitagawa, Y. Ishikawa and A. Morishima. A Rule-oriented Architecture to Incorporate Dissemination-based Information Delivery into Information Integration Environments. Proc. 2000 ADBIS-DASFAA Symposium on Advances in Databases and Information Systems, Sep. 2000.
- [3] N. W. Paton and O. Diaz. *Active Database Systems*. ACM CS, March, 1999.
- [4] T. Kajino, H. Kitagawa and Y. Ishikawa. Algebraic Service Specification and Rule Generation for Integrating Multiple Dissemination-Based Information Sources. Proc. the 7th International Conference on Database Systems for Advanced Applications(DASFAA 2001). 2001(To appear).