様式Ｃ－１９

科学研究費助成事業（科学研究費補助金）研究成果報告書

平成２４年４月４日現在

機関番号：12102
研究種目：若手研究(B)
研究期間：2010～2011
課題番号：22700090
研究課題名（和文）高機能演算子を有する分散実時間ストリーム処理基盤に関する研究

研究課題名（英文）A Study on Distributed Real-Time Stream Processing Infrastructure with Advanced Operators
研究代表者
川島　英之（KAWASHIMA HIDEYUKI）
筑波大学・システム情報系・講師
研究者番号：90407148

研究成果の概要（和文）：本研究では高機能演算子を有する分散実時間ストリーム処理基盤の研究開発に取り組んだ．高機能演算子としてはベイジアンネットワーク，そしてメディアデータ管理を実現した．その他に高速永続化機構，高信頼化機構，暗号化ストリーム処理，そしてストリーム結合処理に関する研究を行った．

研究成果の概要（英文）：This study tried to develop a distributed real-time stream processing infrastructure with advanced operators. As for such operators, Bayesian neworks and media data management are developed. In addition, I conducted research on fast data persisting, high availability scheme, encrypted stream processing, and stream join operators.

交付決定額

（金額単位：円）

	直接経費	間接経費	合　計
2010 年度	1,800,000	540,000	2,340,000
2011 年度	1,300,000	390,000	1,690,000
年度			
年度			
年度			
総　計	3,100,000	930,000	4,030,000

研究分野：　ストリームデータ処理
科研費の分科・細目：メディア情報学・データベース
キーワード：ストリームデータ処理

１．研究開始当初の背景

実世界で生じるイベントを検知すべく，屋内外のセンサデータを収集・解析する取り組みが行われていた．この状況に対して，センサデータ処理基盤に関する研究が行われていた．国外研究には，信号処理に特化した WaveScope(MIT)，ノイズ除去を行う MauveDB(MIT)，複合イベント処理（CEP）を対象とする SASE/CLARO（マサチューセッツ州立大），関係ストリーム処理を支援する Borealis（MIT）が挙げられる．国内研究には，KRAFT（提案者）と異種情報統合基盤 StreamSpinner(筑波大-北川研)があった．

２．研究の目的

本研究の目的は，上記のような個別システムを開発することではなく，統合的なシステムを実現することであった．そのためにリレーショナルデータモデルを基礎とし，リレーショナルデータ処理，complex event processing，そしてベイジアンネットワークなどの推論処理，そしてメディア管理機能など，様々な機能を統合的に扱えるシステムを開発することが研究目的であった．

３．研究の方法

本研究では，まず個別技術を開発し，後にそれらを統合してシステム化する，という方法を採用した．これにより，研究用プロトタイプシステムと実用システムを明確に切り分けることができ，研究と開発の速度を上げることができたと考える．

４．研究成果

本研究では高機能演算子を有する分散実時間ストリーム処理基盤の研究開発に取り組んだ．高機能演算子としてはベイジアンネットワーク，そしてメディアデータ管理を実現した．その他に高速永続化機構，高信頼化機構，暗号化ストリーム処理，そしてストリーム結合処理に関する研究を行った．ここでは特に暗号化ストリーム処理について記述する．本研究を発展させ，トップ会議を狙う．

近年のセンシングデバイスの発達に伴い，継続的にデータを生成し続けるストリーム情報源の数が増大しつつある．ストリーム情報源にはカメラ，ルータを流れるパケット，株価・為替変動などの金融情報，GPS 端末をはじめとする各種センシングデバイスなどが挙げられる．これらの情報源から得られるデータをストリームデータと呼び，ストリームデータを処理する基盤システムとして，ストリーム処理エンジン（SPE: Stream Processing Engine）がこれまで開発されてきた．ストリーム処理エンジンとは，ストリーム情報源から無限に到着するストリームデータに対して連続的問合せを行うことを可能にするシステムである．連続的問合せとして，選択・射影・結合・集約などのリレーショナル演算や複合イベント処理を行うことができる．この問合せは SQL ライクの言語を用いて記述することができ，データが到着する度に評価が行われる．また，メモリ上で即時的に実行されるため，データ永続化を必要とする従来のデータベース管理システム（DBMS）に比べ，処理を高速化できる．また，ストリームデータ処理などのビックデータ処理を実現するための資源提供技術として，近年パブリッククラウドが注目を浴びつつある．パブリッククラウドとは，インターネットを介して不特定多数の個人または企業に対して計算資源を提供するサービスを指す．ユーザーは，自前で計算資源を持つ場合に比べて運用・維持コストの削減や，資源の柔軟な増減が可能になる．パブリッククラウドの一例として，米 Amazon Web Services による Amazon EC2 ，米 Microsoft による Windows Azure などが挙げられる．しかし企業は，これらパブリッククラウドに対してセキュリティに関する懸念を抱いている．パブリッククラウドは一般に，企業のファイヤウォールの外側で第三者により管理される．そのため，パブリッククラウド上に保存された情報を，管理者を含む第三者よって覗き見られたり改ざんされる可能性がある．この問題に対する解決策の一例として，近年，DBMS 上で暗号化されたデータに対してリレーショナル演算を可能とするための研究が行われてきた．

以上を踏まえ我々は，安全性を考慮したストリームデータ処理の実現を目的とし，本稿で暗号化ストリーム処理方式の実現，及び通信帯域・メモリ使用量の効率化を実現する手法を提案した．

提案手法は CryptDB という技術に基づく．CryptDB は，暗号化された値に対してリレーショナルデータベース上でリレーショナル演算を可能とする手法である．一方で我々の研究では，暗号化された値に対して SPE 上でストリームデータ処理を行う．一般に，リレーショナルデータベースではディスクに保存されているデータに対して問合せを行うが，SPE では逐次的に到着するデータに対し，オンメモリで連続的に問合せを行う．そのため，ストリームデータ処理に CryptDB の考え方を適用するためには，ストリーム情報源と SPE との間に，ストリーム情報源から発信された情報を逐次的に暗号化するためのモジュールが必要となる．我々はこれを Encryption Module と呼ぶ．Encryption Module は，ストリーム情報源から逐次的に発信される情報を受け取り，属性毎に３種類の Onion(Onion Eq, Onion Ord, Onion Add)と１つの初期化ベクトル(IV: Initial Vector)を生成して SPE へ送信する．

パブリッククラウドにおいてストリームデータ処理を行う際，ストリーム情報源が広範囲に分布していることが考えられる．この時，Encryption Module はそれぞれの信頼された領域(Trusted Area) 内に設置され，パブリッククラウドを含むインターネット(Untrusted Area) 上には暗号化されたデータのみが流通するものとする．また，クエリ結果を受け取るユーザの存在する領域には Decryption Module を設置し，クエリ結果の復号や，SPE 上で暗号値に対して実行することのできなかったクエリの後処理(Post-Processing) を行う．

この仕組みを実現した後，通信帯域とメモリ量を削減する技術を開発した．まず，Encryption Module において暗号化によるデータ量の増加を抑え，データ転送量を削減するための手法を述べる．事前に登録されたクエリを解析する事で，クエリ処理に必要な Onion の種類を判断することができる．必要な Onion の種類を事前に Encryption Module に告知することで，Encryption Module は必要最小限の Onion のみを生成し，暗号化に要

するコストの削減と，EncryptionModule から SPE へ転送されるデータ量の削減をすることが可能となると考えられる．

前述した提案手法では，Encryption Layer 及び EncryptionLayer から SPE に入力されるまでの経路において暗号化コストとデータ量を削減する手法について述べた．提案手法 II では，SPE 内部でのメモリ使用量を削減するための手法を提案する．SPE に複数のクエリが登録されている場合，すなわち処理木が非線形となる場合には，提案手法 I に加え，処理木(Processing-tree) 中の各演算子毎に入力される Onion の種類を必要最小限にすることで，SPE におけるメモリ使用量の削減が可能となると考えられる．特にウインドウ結合(Window-Join)や集約演算など，シノプシスにタプルを保存する必要のある演算子おいて，その後の処理に必要のない Onion を含んだタプルを保存することはメモリ使用量の浪費につながる．そこで，これらの演算子の前に射影演算子を挿入し，その後の処理に必要のない Onion を適宜取り除いてゆくことで，SPE のメモリ使用量の削減を図った．

上で提案した各提案手法について，プロトタイプシステム作成し評価実験を行った．実験を行うにあたり作成したプロトタイプシステムは，Java 言語を用いて実装を行った．温度センサ及び湿度センサを仮定したモジュールから乱数値を発生させ，Encryption Module で暗号化処理を行う．ここで，温度センサと湿度センサからのストリームデータは交互に到着するものとした．暗号化アルゴリズムには，DET として CBC モードの AES 暗号を，また HOM として Paillier 暗号 を用いた．これらの暗号化処理には，既存のライブラリを用いた．OPE に相当する暗号化処理を提供するライブラリは調査した限りでは存在しなかったため，実装を今後の課題とし，本実験では Encryption Module における Onion Ord の作成は割愛した．

はじめに，データ転送量削減手法を適用した場合と適用しない場合とで，Encryption Module における暗号化コストの比較，及び SPE に転送されるデータ量の比較を行った．処理木における結合演算は，ウインドウサイズが 1 のウインドウ結合とした．本実験における例では，温度センサから発せられる情報{id,temp} の各属性に対して Onion Eq 及び Onion Add を生成し，出力される情報は{id-Eq, id-Add, temp-Eq, temp-Add} の属性を持つ．同様に，湿度センサから発せられる情報{id, humid}の各属性に対しても Onion Eq 及び Onion Add を生成し，出力される情報は{id-Eq, id-Add, humid-Eq, humid-Add} の属性を持つ．それぞれのセンサから出力される計 200 タプルに対して，対応する Onion の生成に要する時間と，それらのデータ量を計測した．実験の結果，提案手法はナイーブな手法よりも 2 倍程度高速になることが観測された．

次に，提案手法を適用しなかった場合，2 つの提案手法を両方とも適用した場合のそれぞれにおいて，実行時のメモリ使用量の比較を行った．結合演算子にタプルが 2 つ入力される毎に全シノプシスに保存されているオブジェクトのデータ量を計算し，平均値を 1 試行のメモリ使用量とした．この試行を，結合演算のウインドウサイズを 5 から 70 まで 5 ずつ変化させながら繰り返し行い，比較を行った．いずれの場合も提案手法 I と II を適用した場合がデータ量が最も少なく，提案手法を適用しない場合がデータ量が最も多かった．具体的には，ウインドウサイズに関わらず，提案手法 I のみを適用した場合は適用しなかった場合の 38～40%程度のメモリ使用量削減，提案手法 I と II を適用した場合は 42%程度のメモリ使用量削減となった．本実験では，提案手法 I のみを適用した場合とと提案手法 I と II を両方適用した場合の違いは，処理木内部の集約演算の入力タプルに id 属性の Onion Eq を含んでいるか否かの違いのみである．そのため，本実験では大きな削減効果は得られなかった

５．主な発表論文等

（研究代表者、研究分担者及び連携研究者には下線）

〔雑誌論文〕（計 4 件）

1. 猿渡俊介，川島英之，高木潤一郎，倉田成人，森川博之，“センサデータベースマネージャにおける問合せ処理とデータ圧縮の同時最適化”，情報処理学会論文誌「新たな展開を迎える ITS、モバイル通信とユビキタスコンピューティング」特集号，53(1)，320-335，2012-01-15 http://ci.nii.ac.jp/naid/110008736788 査読有．
2. 三好健文，寺田裕太，川島英之，吉永努，“ストリーム処理エンジン向け動的再構成可能プロセッサアーキテクチャの設計”，情報処理学会論文誌：データベース，Vol. 4，No. 2 (TOD50)，pp. 35 - 51，2011.査読有．https://ipsj.ixsq.nii.ac.jp/ej/index.php?active_action=repository_view_main_item_detail&item_id=74652&item_no=1&page_id=13&block_id=8
3. 三好健文，寺田裕太，川島英之，吉永努，“ウィンドウ結合演算子の FPGA による実現”，電子情報通信学会論文誌．B，通信 J94-B(10)，1313-1322，2011-10-01，査読有．http://ci.nii.ac.jp/naid/110008749646

4. 塩川 浩昭, 北川 博之, 川島 英之, 渡辺 陽介, ″分散ストリーム処理システムにおける高信頼化方式の提案″, 電子情報通信学会論文誌 Vol. J93-D, No. 6, pp. 767-780, Jun. 2010. 査読有.
http://ci.nii.ac.jp/naid/110007618351

〔学会発表〕(計 5 件)

1. Taiga Abe, Hideyuki Kawashima and Hiroyuki Kitagawa, ″An Efficient Stream Archiving Method by Operator Merge and Write Control″, Proc. The 5th. International Workshop on Data Management for Wireless and Pervasive Communications (DMWPC), Fukuoka, Japan, March 26-29. 2012.
2. Yasin Oge, Takefumi Miyoshi, Hideyuki Kawashima and Tsutomu Yoshinaga, ″An Implementation of Handshake Join on FPGA″, Proc. Second International Conference on Networking and Computing (ICNC 2011), Osaka, Japan, pp. 95-104, December 2011.
3. Masafumi Oyamada, Hideyuki Kawashima, and Hiroyuki Kitagawa, ″Efficient Invocation of Transaction Sequences Triggered by Data Streams″ The 2nd International Workshop on Streaming Media Delivery and Management Systems (SMDMS 2011), Proceedings of 3PGCIC, Barcelona, Spain, October 26-28, 2011.
4. Tsubasa Takahashi, Hideyuki Kawashima, and Hiroyuki Kitagawa, ″A Video Manager for Relational Stream Processing Systems″ The 2nd International Workshop on Streaming Media Delivery and Management Systems (SMDMS 2011), Proceedings of 3PGCIC, Barcelona, Spain, October 26-28, 2011.
5. Hideyuki Kawashima, Hiroyuki Kitagawa, and Xin Li, ″Complex Event Processing over Uncertain Data Streams″, Proc. International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC), Fukuoka, Japan, pp. 521-526, November 4-6, 2010.

６．研究組織

(1)研究代表者

川島　英之 (KAWASHIMA HIDEYUKI)
筑波大学・システム情報系・講師
研究者番号：90407148