

【ICDE2010勉強会】

Session 28 : Stream Processing

担当: 渡辺陽介 (東京工業大学)

Continuous Query Evaluation over Distributed Sensor Networks

▶ 著者

▶ O. Jurca, S. Michel, A. Herrmann, K. Aberer (EPFL)

▶ 地理的に分散したストリーム環境における, publish/subscribe システム

- ▶ 複数ユーザの要求に対して条件を満たすデータを配信する
 - ▶ 要求送信コスト, データ転送コストを下げたい
- ▶ 要求間の条件の包含関係を検出し, 重複した処理結果を送信しないように転送経路上のノードにフィルタ処理を配置する
- ▶ ネットワーク全体を把握した中央サーバを仮定しない (ここが重要)



Figure 1

Swiss experiment project

10以上の研究グループ

温度, 積雪, 日照, 地滑り監視

Subscription propagation

Table 1

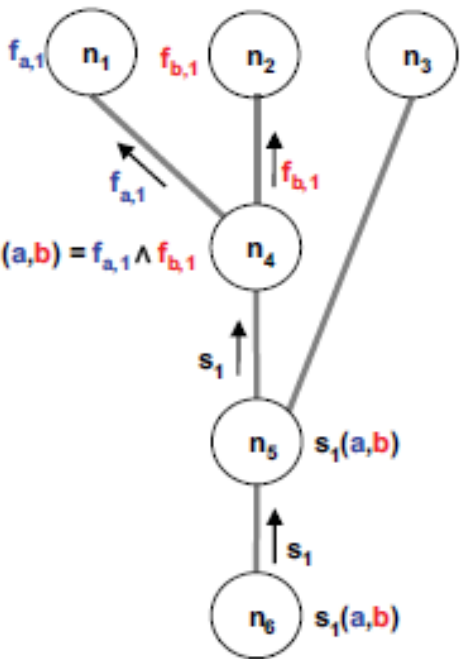
<i>Subscriptions</i>	<i>Sensor a</i>	<i>Sensor b</i>	<i>Sensor c</i>
s_1	$50 < a < 80$	$10 < b < 30$	
s_2		$20 < b < 40$	$2 < c < 20$
s_3	$55 < a < 75$	$15 < b < 35$	$5 < c < 15$

S3はS1とS2の結果から作れる

要求例
(順に登録)

(1) 要求S1を登録

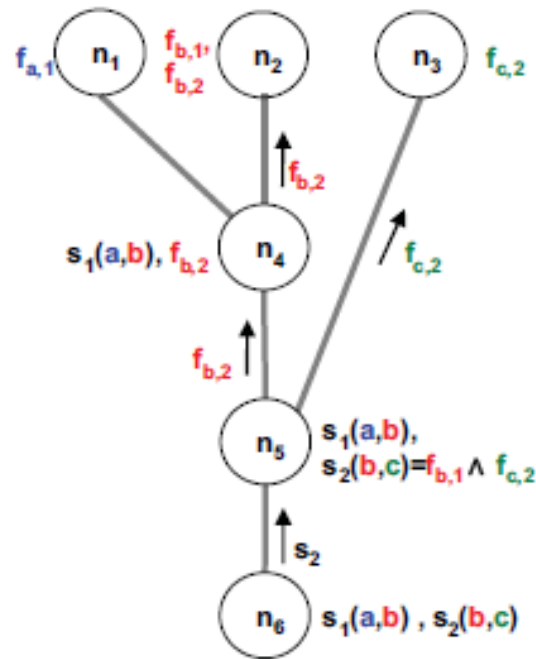
Sensor "a" Sensor "b" Sensor "c"



$$s_1 = \{ (50 < a < 80) \wedge (10 < b < 30) \}$$

(2) 要求S2を登録

Sensor "a" Sensor "b" Sensor "c"

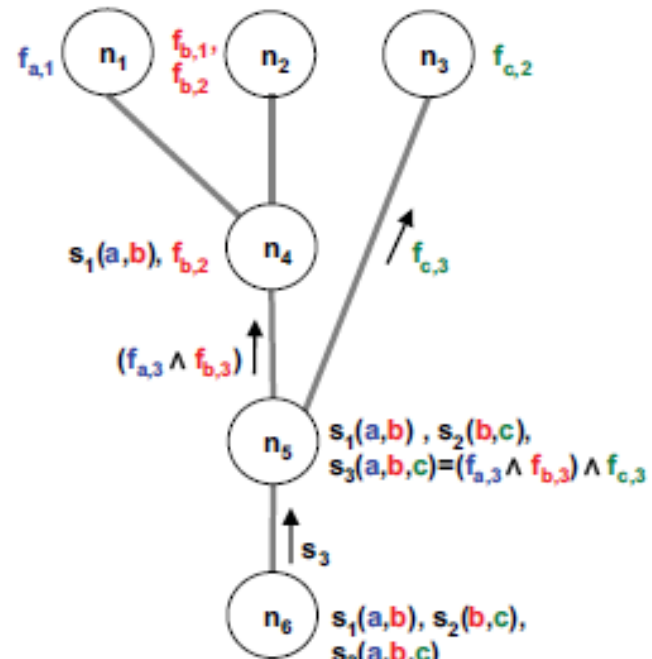


$$s_1 = \{ (50 < a < 80) \wedge (10 < b < 30) \}$$

$$s_2 = \{ (20 < b < 40) \wedge (2 < c < 20) \}$$

(2) 要求S3を登録

Sensor "a" Sensor "b" Sensor "c"



$$s_1 = \{ (50 < a < 80) \wedge (10 < b < 30) \}$$

$$s_2 = \{ (20 < b < 40) \wedge (2 < c < 20) \}$$

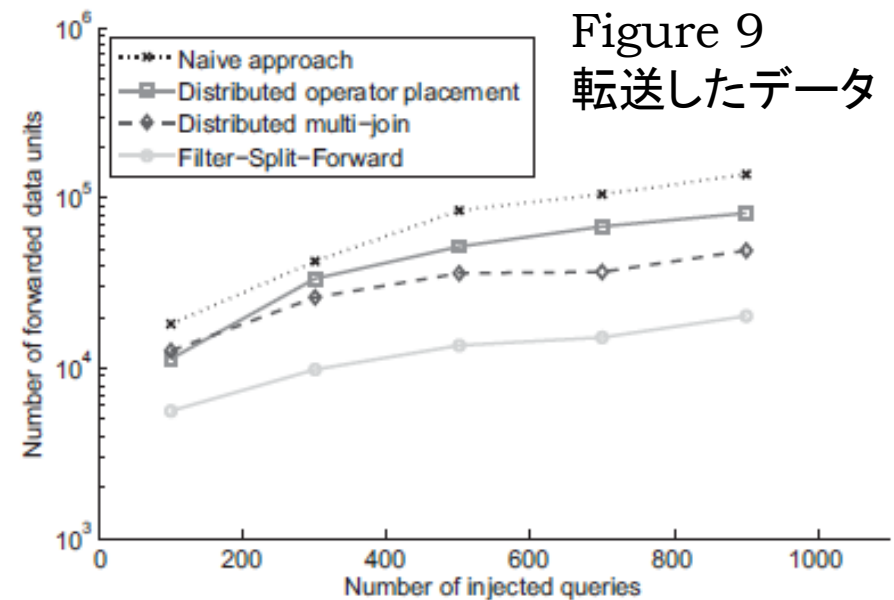
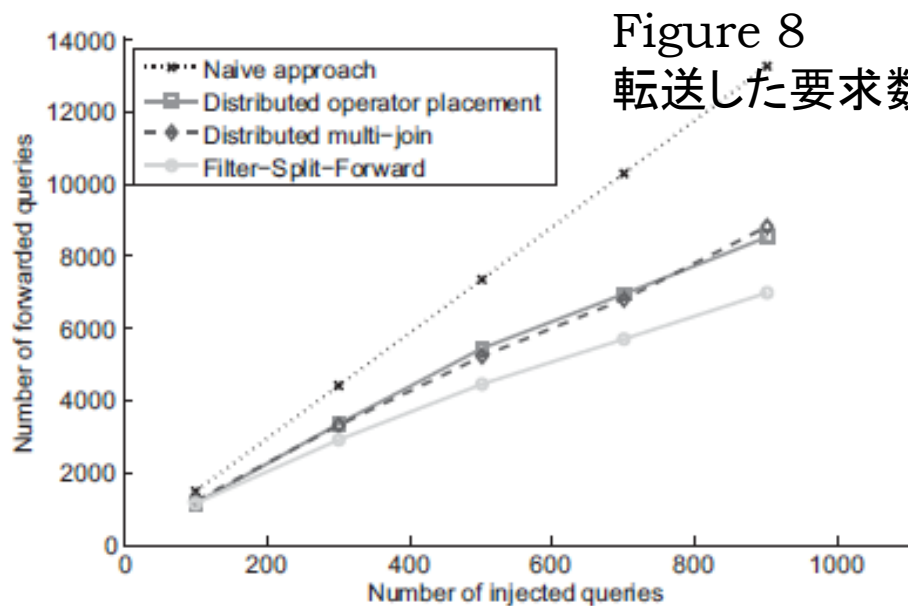
$$s_3 = \{ (55 < a < 75) \wedge (15 < b < 35) \wedge (5 < c < 15) \}$$

Figure 3

Experiments

▶ 比較対象

- ▶ Naïve approach: 各センサーノードへ要求を送る. 共有はしない
- ▶ Distributed operator placement: 共通演算の共有化
- ▶ Distributed multi-join processing: multi-joinをbinary-joinに分解 [7]
- ▶ Filter-Split-Forward (提案手法)



Space-efficient Online Approximation of Time Series Data: Streams, Amnesia, and Out-of-order

▶ 著者

- ▶ S. Gandhi, L. Foschini, S. Suri (UC Santa Barbara)

▶ 時系列データの近似手法

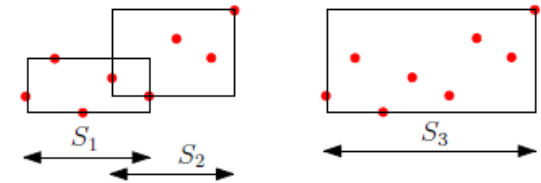
- ▶ 無限にデータが到着するストリームでは、全データを保持することが難しいため、近似による表現を使う

- ▶ Piecewise constant approximation: 代表値で近似

- ▶ Piecewise linear approximation: 線分で近似

▶ Generic-Minmerge (GM) アルゴリズム

- ▶ 通常のデータストリームを対象とした近似手法



▶ Amnesic-Merge (AM) アルゴリズム

- ▶ 古いデータに対しては誤差の許容範囲が広がってよい場合の近似

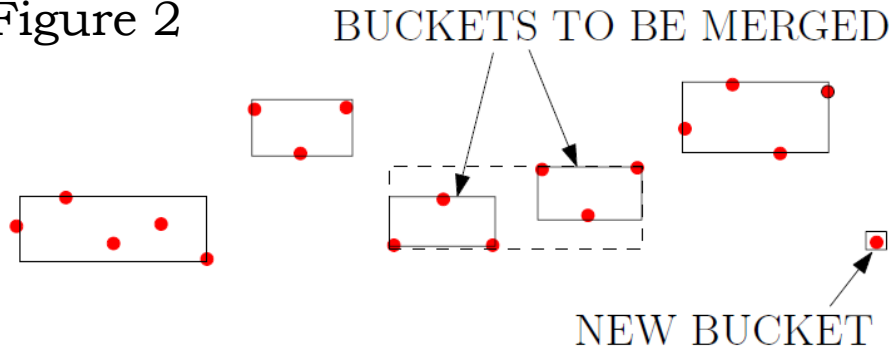
▶ Fragment-Merge (FM) アルゴリズム

- ▶ 到着順序がタイムスタンプ順にならない場合(Out-of-order) の近似

Generic-minmerge アルゴリズム

- ▶ 時系列データを複数のバケットに分割
 - ▶ Piecewise constant approximation: バケット内の点を代表値で近似
 - ▶ Piecewise linear approximation: バケット内の点を線分で近似
- ▶ Generic-minmerge アルゴリズム
 1. ストリームSから新たなデータ v_i が到着する度に, v_i を新しいバケットへ格納
 2. バケット総数が $4B$ 個を超えた場合
 - ▶ 隣り合うバケットのペアから, マージ後の近似誤差が最小になるもの同士をマージ

Figure 2



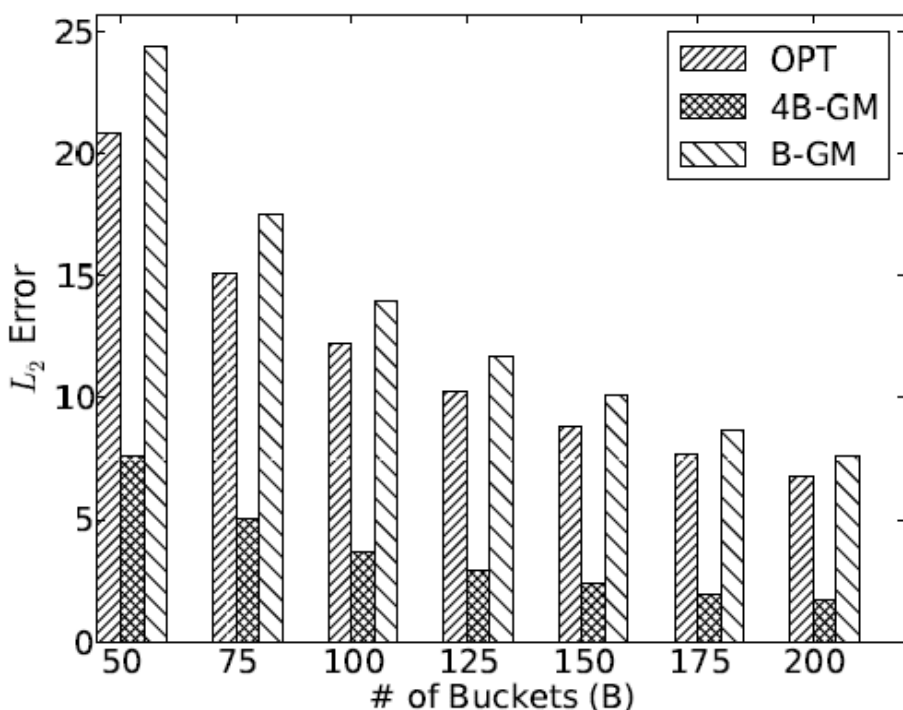
Theorem 4:

B 個のバケットを使う最適な近似法に対して $4B$ 個のバケット数で $\sqrt{2}$ 倍の誤差(L2 norm)を実現

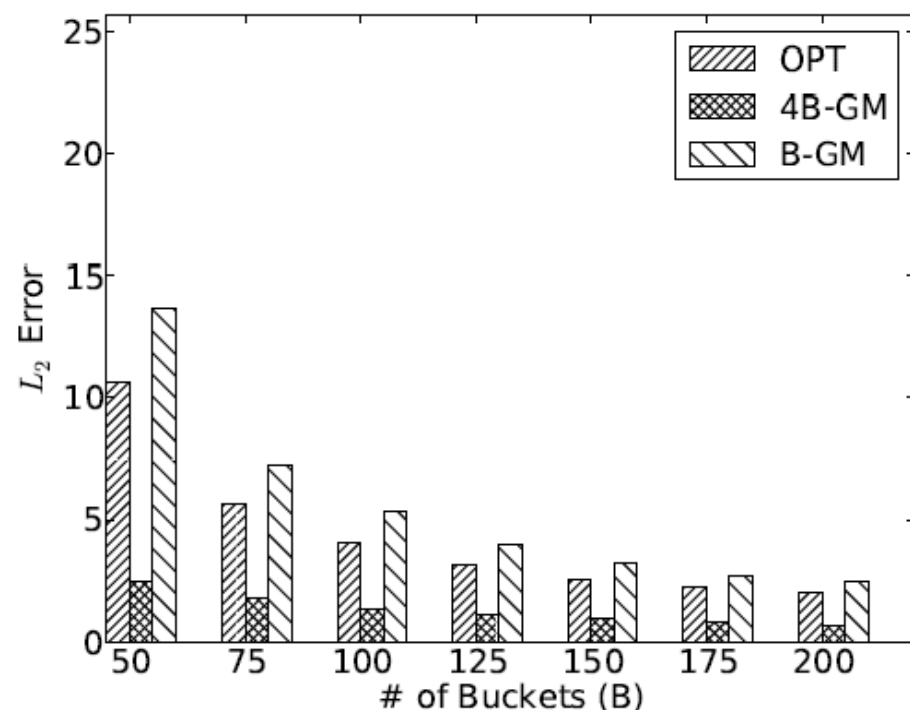
Simulation results (Generic-Minmerge)

実験データ: 4000個の系列データ (湿度センサーのデータ)
比較対象: 動的計画法により求めた最適近似(OPT) [14]

バケット内の点を代表値で近似



バケット内の点を線分で近似



Approximation Trade-Offs in Markovian Stream Processing: An Empirical Study

▶ 著者

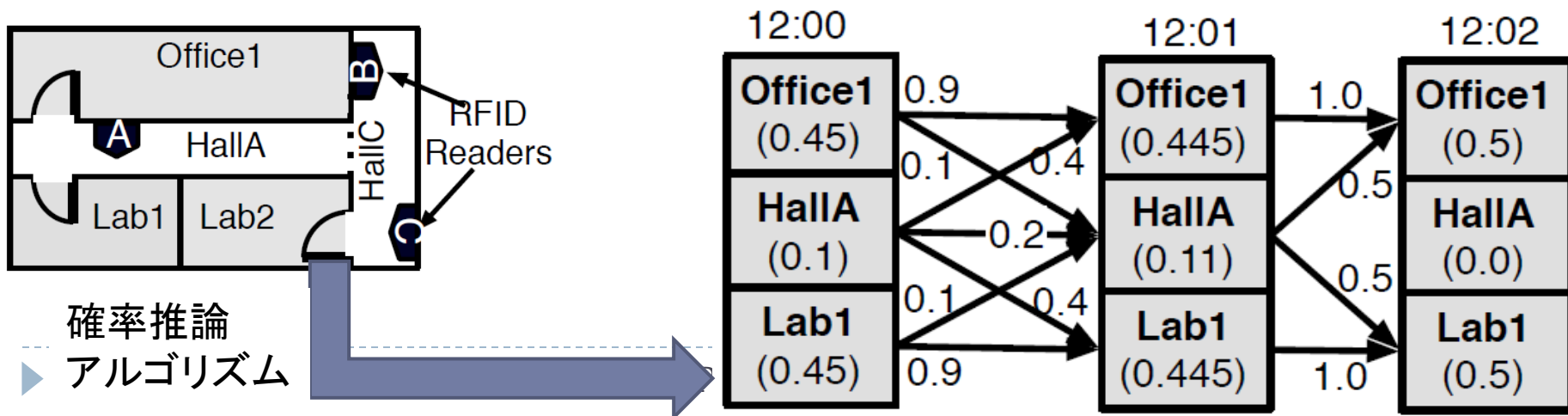
- ▶ J. Letchner (Washington Univ.), C. Re (Wisconsin-Madison Univ.), M. Balazinska (Washington Univ.), M. Philipose (Intel Research)

▶ 確率的なストリームデータにおける近似的な問合せ処理

- ▶ Markovian stream: imprecise and temporary correlated

▶ 問合せ例

- ▶ Event query: ボブが先週コーヒールームに行った時をすべて探せ
- ▶ Boolean aggregation: ボブは月曜日にコーヒールームに行ったか？
- ▶ Count-based aggregation: ボブがコーヒールームに行った回数は何回か？



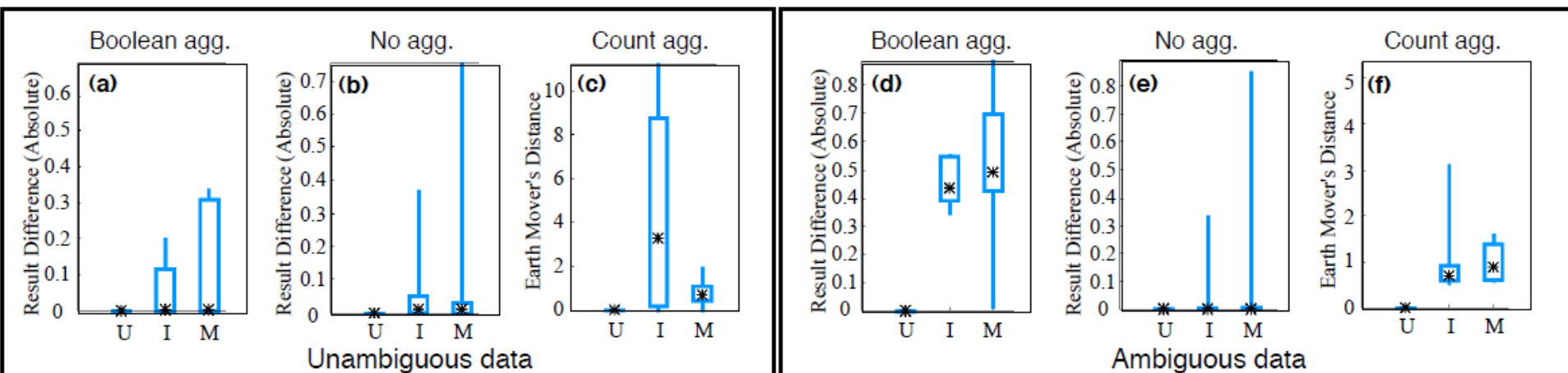
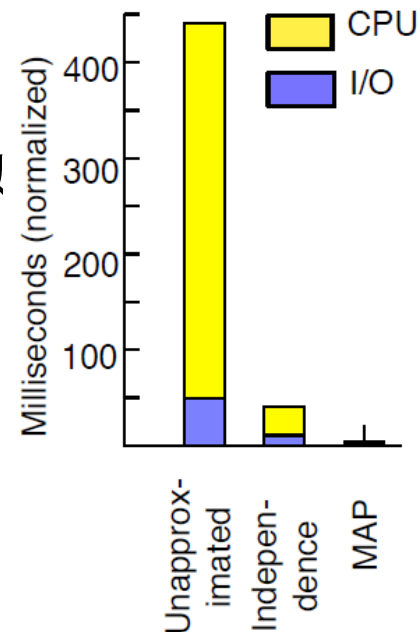
Markovian Stream Approximation

▶ 近似手法

- ▶ MAP (Maximum a Posteriori sequence)
 - ▶ もっとも発生する可能性の高い決定的なシーケンスで近似
- ▶ Independent
 - ▶ Markovian streamにおける時間的な関連を無視する

▶ Empirical study

- ▶ 近似によりパフォーマンスは向上. 問合せの種類によってはMAPがindependentより良いこともある



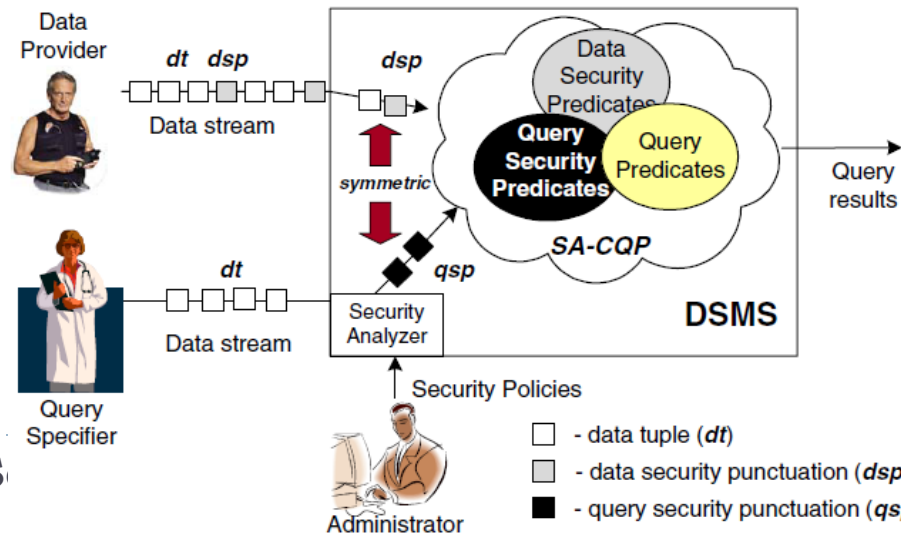
FENCE: Continuous Access Control Enforcement in Dynamic Data Stream Environments

▶ 著者

- ▶ R. Nehme, H. Lim, E. Bertino (Purdue University)

▶ ストリーム環境における動的なアクセス制御

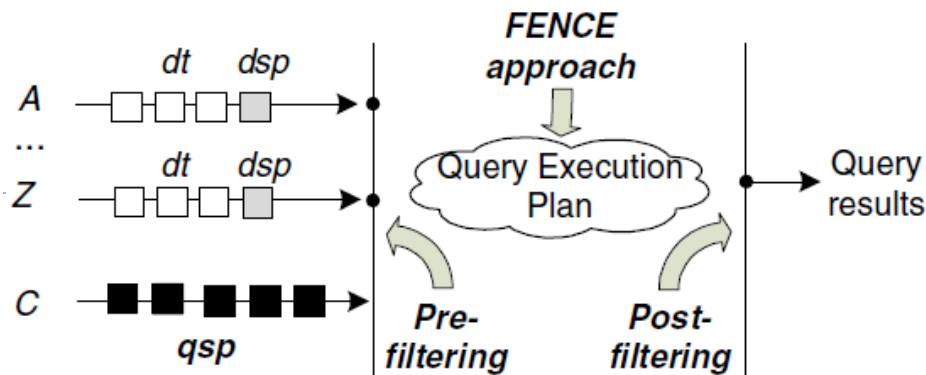
- ▶ ストリームに対する問合せ処理は長期にわたって実行される
 - ▶ 実行中にユーザのデータへのアクセス権限が変化する
 - ▶ “ユーザ1は今からストリーム2のid 30~210のタプルにアクセスしてよい”
- ▶ アクセス権限の変化もストリームデータとして扱う



実現方法

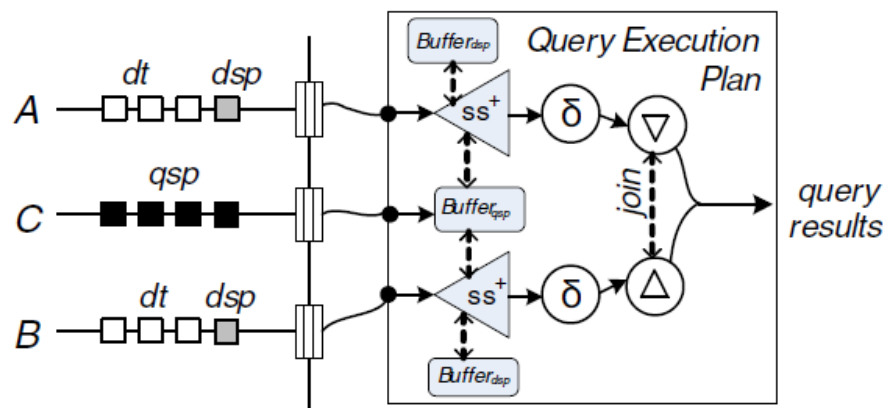
▶ Naïve approach

- ▶ Pre-filtering
- ▶ Post-filtering



▶ Security filter approach

- ▶ 実行プランの中に専用の演算を入れる
 - ▶ Security Shield Plus (SS+)



▶ Query rewrite approach

- ▶ 実行プラン中の選択演算のフィルタ条件を動的に書き換える

