トランザクション基礎研究
バッチ処理への挑戦

2021年10月11日

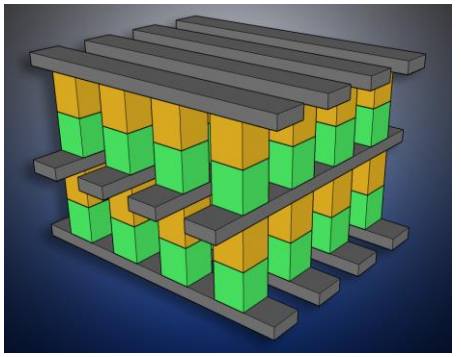
慶應義塾大学環境情報学部
川島英之

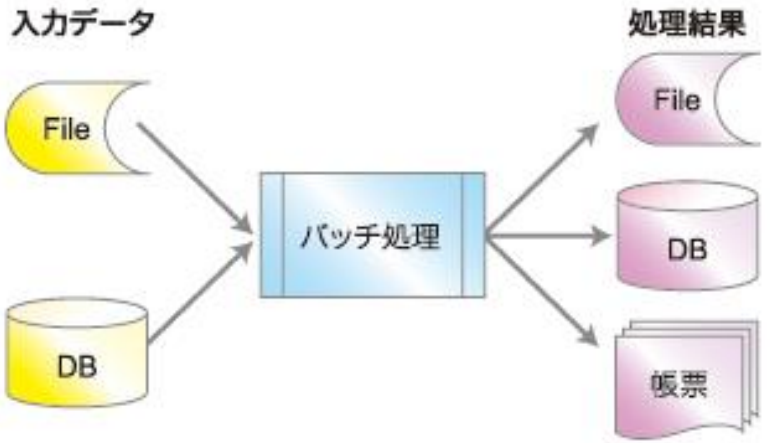
Transaction Today

- 概観
  - 性能：1000万トランザクション/秒以上
  - 技法：楽観法が主流
  - 前提：メニーコア、不揮発メモリ
  - 手法：Silo [1], Cicada [2], etc.
  - 分析器：CCBench [3], DBx1000 [4]
- 既存研究は古いシナリオに依拠
  - Short Transactionのみ
  - 16 operations / TX
  - 全てのサイズは同じ
- 現実のシナリオ
  - **Long Transaction（バッチ）**
  - Real-Time response
  - Dynamic workload

Superdome
(1576 cores)

Non-Volatile Memory
(3D X point)

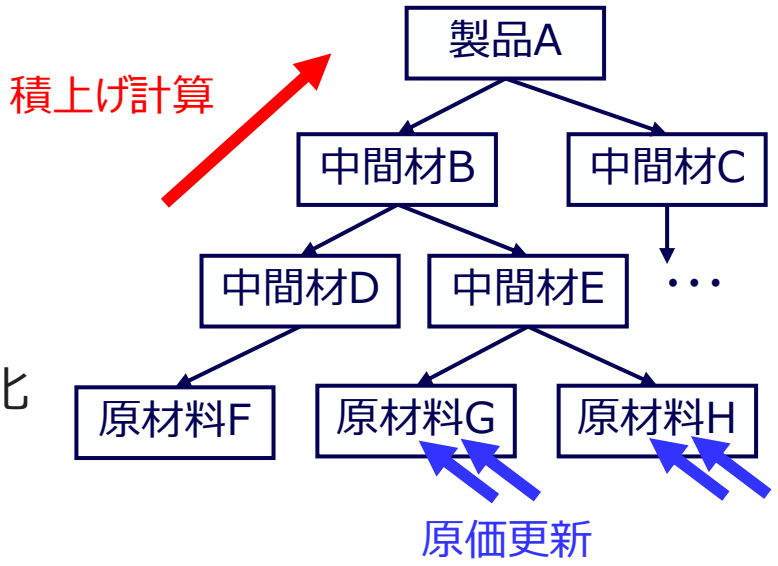




[1] Tu, S., Zheng, W., Kohler, E., Liskov, B. and Madden, S.: Speedy transactions in multicore in-memory databases, *SOSP* (2013).
[2] H. Lim, M. Kaminsky, and D. G. Andersen, “Cicada: Dependably fast multi-core in-memory transactions,” *SIGMOD* (2017).
[3] Tanabe, T., Hoshino, T., Kawashima, H. and Tatebe, O.: An Analysis of Concurrency Control Protocols for In-Memory Databases with CCBench, *PVLDB* (2020).
[4] DBx1000: A single node OLTP database management system. https://github.com/yxymit/DBx1000, 2016

現実的なワークロード:バッチ処理とオンライン処理が混在

- **例1) 製造業におけるBOM(Bill Of Materials)ベースの製造原価計算**
  - バッチ処理: 対象品目を構成する材料を再帰的に取得し、原価を積上げ計算
  - オンライン処理: 材料原価更新、品目構成更新
- **例2) 通信業におけるオンラインビリング**
  - バッチ処理: 通話料金計算
  - オンライン処理: 契約マスタ更新、通話履歴更新
- **どちらもバッチ処理の参照対象が頻繁に更新**
  - 悲観法/2PL: デッドロック or オンライン処理が長時間化
  - 楽観法/OCC: バッチ処理が毎回アボート



例1) 製造業におけるBOMと原価計算

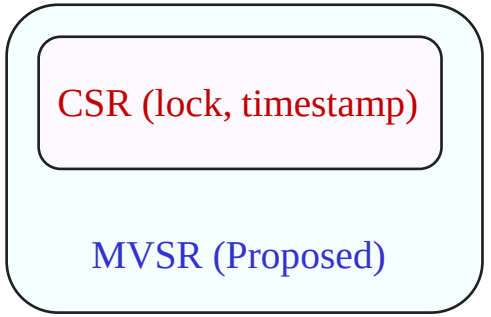
既存の並行性制御手法の課題

- 現在の解決策: ユーザに制約を強いて対応
  - オンライン処理を止めてバッチ処理を実施
  - スナップショットやレプリカを作ってバッチ処理を実施
  - デッドロックしないようにアプリケーションが頑張って制御

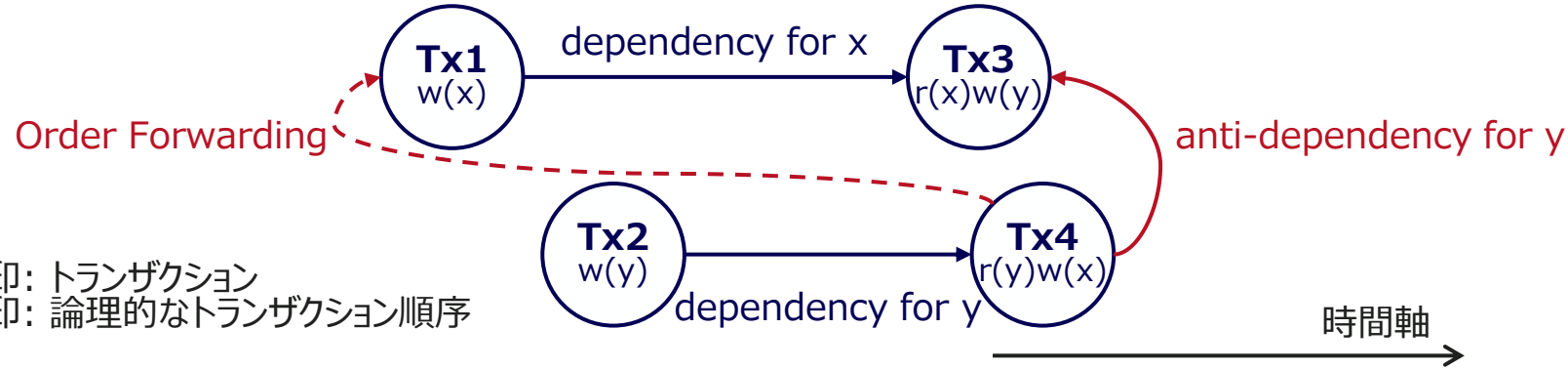
- Tsurugi研究における提案: DBMS側で対応
  - 長いバッチ処理と短いオンライン処理が混在するワークロードを通せる
    新しい並行性制御プロトコル(Oze/尾瀬)を考案、プロトタイプ実装
  - 現在、評価・論文執筆中
  - 将来的にはTsurugiのSiloベースプロトコルに加え、選択肢の1つになることを目指す

プロトコル Oze 概要

- マルチバージョンの並行性制御プロトコル
  - 論文で主流のロックベースやタイムスタンプベースではない
  - Serializable
- 中央集権的なデータ構造を用いずに各トランザクションの順序関係を管理
- *MVSR(Multiversion View Serializability)*空間で動作
  - PostgreSQL, MySQL >
  - SSI, Silo… >
- スケジュール空間を広げる order forwarding の提供
  - 他トランザクションが読んだものを壊さない範囲で論理的なWrite順序を変更



Scheduling Space



近代的研究

Method	Year	Conference	Features
CICADA	2017	SIGMOD	Optimistic Multi-Version
MOCC	2016	VLDB	Optimistic Pessimistic
TicToc	2016	SIGMOD	Optimistic
ERMIA	2016	SIGMOD	Multi-Version
Silo	2013	SOSP	Optimistic
SI	1995	SIGMOD	Multi-Version

まとめ

- 現実のトランザクションはバッチとオンラインが混在
- 従来研究はいずれも未対応
- 新プロトコルOzeの提案と評価 (CCBench)
- 今後の予定
  - Tsurugi導入（課題: Silo/OCCとの統合、効率的GC，分散環境適用）
  - 論文とコード公開予定

Takayuki Tanabe, Takashi Hoshino, Hideyuki Kawashima, Osamu Tatebe: An Analysis of Concurrency Control Protocols for In-Memory Database with CCBench. PVLDB, 2020.