管理者に対しても機密を保持できる暗号化データベースの索引構成法

三浦 志保　(指導教員：渡辺 知恵美)

1　はじめに

近年，データベース製品の管理運用を外部のデータベース技術者に委託するデータベースアウトソーシングサービスが普及しつつある．このときサービスの利用者は，委託業者の管理者に機密事項であるデータの内容を閲覧されたくないという要求を持つ．そこで，利用者側が管理者に頼ることなく情報漏洩を防ぐことのできる暗号化データベースシステムの研究が進められている [1]．このシステムでは，暗号化したデータベースと索引がサーバに置かれる．本研究では，タップルの内容を侵入者に推測されにくいより安全な索引構成法を提案する．

2　暗号化データベースシステム

2.1　暗号化データベースに対する処理の流れ

図 1 に暗号化データベースに対する処理の流れを示す．データは全てクライアント側で暗号化し，セキュアな索引を付与してから，サーバ側の暗号化データベースに格納される．これで例えデータベース管理者であっても，タップルの内容を知ることはできなくなる（図 1①）．検索は 2 段階問合せにより実現する（図 1②）．2 段階問合せとは，大まかな結果候補を抽出するためサーバ側の索引に対して行う filtering query と，結果候補に対して解精製処理を行い最終的な結果を求める refinement query とから成る．

またサーバ側には，暗号化データベースの他に索引情報（図 1 ★）を置き，クライアントがデータを挿入・更新したり検索したりする際には，自動的にここから情報を取得できるようにする．

2.2　暗号化と索引の付与

暗号化データベースのテーブルは，元データのタップルを暗号化した文字列（etuple）と各属性の索引（属性^S）とで構成されている．社員というテーブルで例を図 2 に示す．まず，社員 (社員番号, 氏名, 住所, 月給) ⇒ 社員^S($etuple$, 社員番号^S, 氏名^S, 住所^S, 月給^S) とし，社員^S をサーバ側の暗号化データベースに格納する．1 行目の etuple には “101101101”が格納されているが，これはタップル “150，浅賀千里，埼玉県...，460000” を暗号化したものである．索引について月給を例に考えると，まず，ドメイン領域 (0, 100 万] を 20 万ずつ 5 つの領域に分割する．以下，分割されたそれぞれの領域のことをバケットと呼ぶ．そして，各バケットにランダムにバケット番号を与える．したがって，月給 =460000 は 月給^S = 12587 として格納される．

また索引情報には，テーブル名，属性，バケットの下限値・上限値，バケット番号，バケット内のタップル数が暗号化された文字列となって格納される．

2.3　検索

ユーザが発行した問合せ文は，以下の 2 段階の問合せにより実行される．例えばクライアントが，月給が 55 万円以上の社員の名前を求めるために「SELECT

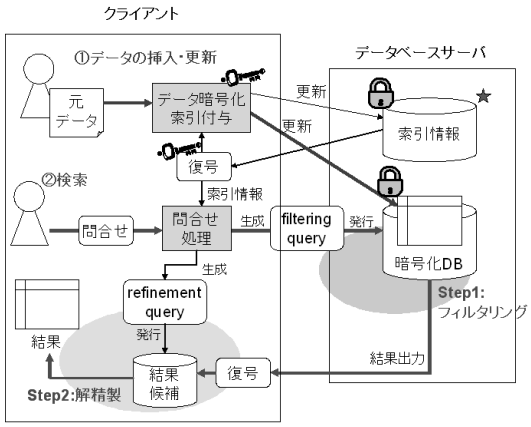

図 1: 暗号化データベースに対する処理の流れ

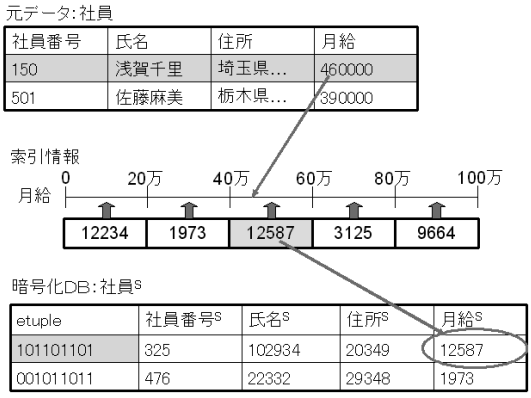

図 2: 暗号化と索引の付与

名前 FROM 社員 WHERE 月給 ≥ 550000」を発行したとする．その場合，まずはサーバから索引情報を自動取得してきて復号化を行い，その情報を元に filtering query 「SELECT etuple FROM 社員^S WHERE (月給^S = 12587 or 月給^S = 3125 or 月給^S = 9664)」を発行する．この処理により月給が 40 万円以上の社員のタップルが得られる．この結果候補をクライアント側で復号化した後に，refinement query 「SELECT 名前 FROM 社員 WHERE 月給 ≥ 550000」を発行し，最終的な結果を得る．

この仕組みでは，サーバ側でフィルタリング処理をすることによりクライアント側での処理が少なくてすむため，検索効率をさほど下げることなく安全性を保つことができる．

3　安全な索引構成法の提案

3.1　従来の索引構成法による安全性

文献 [1] で用いられている等間隔の領域分割法では，元データにおいて同じ値（または近い値）同士のものは同じバケットに入るため，暗号化データベース内でも同じ値で格納される．さらに，それぞれのバケットに入っているタップル数は一目で分かる．したがって，それぞれのバケットに入っているタップル数に偏りがある場合，分布解析などによりデータの内容を予測されてしまう可能性がある．本稿では，この問題を解決

する索引構成法を提案する．

3.2 素朴法の提案

3.2.1 基本概念

本研究では，MaxDiff[2] での領域分割法を用いる．具体的には，データが挿入される度にそれぞれのバケットの範囲を変えることにより，各バケットの該当タップル数に一定数以上の差を生じさせないようにする．以下，各バケットの該当タップル数の差の上限を MaxDiff と表す．バケットを分割する方法はいくつか考えられるが，本稿ではバケットの個数の上限を決めておく方法について述べる．

3.2.2 バケット調整法

クライアントがデータ挿入したとき，まずはサーバから索引情報を取得し復号化する．その情報を用いて，必要ならばバケット調整を行い，索引情報および暗号化データベースを更新する．

バケット調整の方法を図 3 の場合を例に説明する．ただし各バケット p_j は (下限値, 上限値] と表し，バケット数の上限は 5，MaxDiff は 2 とする．今，バケット p_3 と p_5 との該当タップル数の差は 3 であり，MaxDiff を超えているのでバケット調整が必要である．該当タップル数最大のバケット p_3 から隣のバケット p_2 へ，タップルの切り崩しを行う．p_3 内の最小値である 53 を p_2 の上限値および p_3 の下限値とする．その結果，p_2 の該当タップル数が最大になり，最小該当タップル数との差が MaxDiff 内におさまっていないので p_2 に対して同様に切り崩し処理を行う．このようにして切り崩し処理を繰り返してもその差が MaxDiff 以下にならない場合は，該当タップル数が最小であるバケットに対して隣のバケットからタップルを埋め合わせする．バケットの埋め合わせ処理を行っても各バケットの該当タップル数の差が MaxDiff 以上ある場合は該当タップル数が最小であるバケットにダミータップルを挿入することで調整をはかる．詳しいアルゴリズムは論文 [3] に示しているので参照されたい．

3.3 ヒストグラムを導入した改良法の提案

3.3.1 基本概念とバケット調整法

素朴法では，バケット調整の際に，切り崩しを行うバケットのタップルデータを全て暗号化テーブルから取得し復号化しなければならない．そのため，サーバから取得するタップル数に連動して実行時間も増加してしまう．そこで，サーバから取得するデータの量を最小限に抑えるために，それぞれのバケット内を等間隔の bin に分割し，そのヒストグラムを索引情報に付加する．切り崩し処理で新しい境界線を決定する際，このヒストグラムを利用する．これによりサーバからデータを取得する必要がなくなり，索引情報を見るだけでバケット調整ができる．また 1 個のデータ挿入の際に行われるバケット調整が複数回ある場合は，調整の度にヒストグラム情報を含む索引情報だけを更新し，バケットの範囲変更の情報は蓄積していく．そして全ての調整が終了してから暗号化テーブルを更新する．これにより，更新タップル数も最小限に抑えられる．

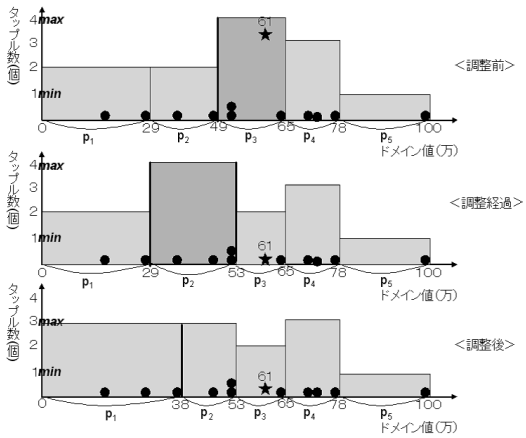

図 3: バケットの切り崩し

3.3.2 改良法の検証

素朴法と改良法とで同じ条件でデータ挿入を行い，その実行時間を計測した．ただし，バケット数の上限は 5，MaxDiff は 2，bin の幅は 20 とした．図 4 から最大実行時間が 3 分の 1 程度になっていることが分かる．ヒストグラム導入は実行時間短縮に有用であると考えられる．

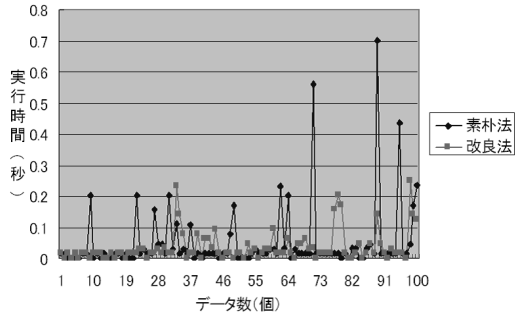

図 4: 実行時間の比較

4 まとめと今後の課題

本稿では新しい索引構成法の 1 つとして，MaxDiff での領域分割法を提案した．また，ヒストグラムを導入し計算量を小さくする方法も併せて提案した．

今後は削除や更新に伴うバケット調整，適切なバケット数やバケット内タップル数の検証やそれらの動的変更，複数の属性への対応，部分一致による検索などを実装・検証していきたい．

参考文献

[1] H. Hacigumus, B. Iyer, C. Li, and S. Mehrotra.: “Executing SQL over Encrypted Data in the Database-Service-Provider Model,” In *Proceeding of the 2002 ACM SIGMOD International Conference on Management of Data*, pp. 216–227, June 2002.

[2] V. Poosala, P. J. Haas, Y. E. Ioannidis and E. J. Shekita.: “Improved histograms for selectivity estimation of range predicates,” *Proceedings of ACM SIGMOD*, pp.294-305, 1996.

[3] 三浦志保, 渡辺知恵美.: “管理者に対しても機密を保持できる暗号化データベースの索引構成法,” 第 18 回データ工学ワークショップ, 2007,（投稿中）.