

## 組込み用データベースにおける検索方式に関する一考察

氏家 純也<sup>†</sup> 水口 武尚<sup>†</sup> 大塚 義浩<sup>†</sup> 田中 功一<sup>†</sup>三菱電機 (株) 情報技術総合研究所<sup>†</sup>

## 1. はじめに

近年の組込み機器で扱うコンテンツの肥大化に伴い、DBMS (Database Management System) を用いたコンテンツの管理を採用するケースが増えている。また、組込み機器ではユーザ操作に対する応答時間が製品の品質を測る上で 1 つの重要なファクタである。したがって、組込み機器向け DBMS を用いた検索に着目した場合、DBMS 単体の応答性能だけでなく、ユーザインタフェースと組み合わせた場合の応答性能の向上が課題であると言える。

本稿では、ユーザ操作に基づきアプリケーションから発行される検索要求を複数の検索要求に変換して DBMS に発行することでユーザ操作に対する応答時間の短縮を実現する方式を提案する。

## 2. 組込み用 DB に求められる機能

組込み機器で扱うコンテンツの種類はテキストデータだけでなく、静止画、動画、音声など増加の傾向にあり、コンテンツを蓄積する不揮発メモリのサイズも数十 MB、多いものでは数 GB まで増えている。これらのコンテンツの管理に DBMS を用いることで、コンテンツを効率的に扱えるだけでなく、データ管理機能をアプリケーションから分離し DBMS に集約することによる開発コストの削減、またはフットプリントの削減といった効果が期待できる。

コンテンツ数の増加に伴い、コンテンツの検索にかかる時間も増加してしまう。しかし、組込み機器ではユーザ操作に対する応答時間が重要である。また、組込み機器は表示装置 (ディスプレイなど) のサイズが小さいものが多く、検索結果が複数の画面 (ページ) に分割されるケースが多い。したがって、組込み用 DB の検索機能に対する課題は「組込み機器の表示装置に表示する件数の検索結果を短時間で DB 利用者 (アプリケーションなど) に提供すること」と言える。

次に、ユーザによる表示するページの変更に

A Study of Retrieval Method for Embedded Database  
Jun'ya Ujiie<sup>†</sup>, Takehisa Mizuguchi<sup>†</sup>, Yoshihiro Otsuka<sup>†</sup>, and  
Koichi Tanaka<sup>†</sup>

<sup>†</sup>Information Technology R&D Center, Mitsubishi Electric Corporation

着目すると、この機能には連続的な前後ページへの移動だけでなく、「10 ページ前/後」または「最終ページ」といった不連続なページ移動の実現も求められると考える。これらの機能を実現するためには、アプリケーションが検索結果として得られるレコードの数 (ヒット件数) を取得する必要がある。しかし、検索結果を得る前にヒット件数のみを取得することは、同じ条件の検索を繰り返すことになるため、ユーザ操作に対する応答時間の増加につながる。

また、アプリケーションから指定された位置 (例えば 3 件目) にあるレコード (検索結果の 1 行に相当するデータ) を DBMS が提供するためには、DBMS が指定されたレコードをデータベースファイルから取得しなければならない。しかし、DBMS も検索結果先頭から件数を数えることにより指定された位置にあるレコードを取得するため、先頭ページの表示と比べて応答時間が長くなってしまおうといった問題がある。

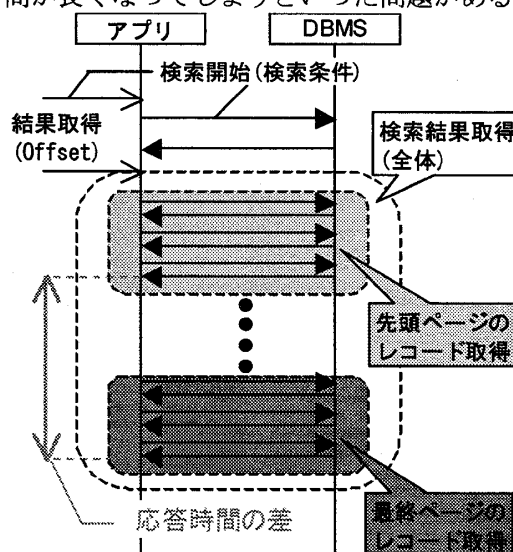


図 1 表示位置の違いによる応答時間の差

図 1 はユーザ操作によりアプリケーション (図中では「アプリ」) が DBMS に検索開始を要求し、次のユーザ操作により「Offset」で特定されるレコードを表示する動作を表したシーケンス図である。アプリケーションは検索結果を先頭から順番に取得しているため、表示対象の検索結果における位置を意味する「Offset」が検索結果の後方を示す場合、検索結果の前方にある検索結

果より取得に時間がかかってしまうことが分かる。

### 3. 提案手法

本稿で提案する検索方式について述べる。提案手法は、テーブルが持つ任意のレコードをレコード番号（プライマリー）で一意に特定可能であることを前提とする。

提案手法では、アプリケーションより発行された検索要求を以下の2種類の検索要求に変換し、DBMSに検索要求を発行する。

- (1) 検索条件を満たすレコード番号を取得する検索
- (2) レコード番号を条件としてカラムの値を取得する検索

(1)の検索条件、および(2)で取得するカラムの一覧はアプリケーションから与えられた検索要求を用いる。例えば、アプリケーションから渡された検索要求が「SELECT a FROM t WHERE b=10」というSQL文の場合、(1)では「SELECT RowId FROM t WHERE b=10」、(2)では「SELECT a FROM t WHERE RowId=?」という検索が実行される。なお「RowId」はプライマリーであり、「?」には(1)にて取得したレコード番号が設定される（レコード番号を複数取得した場合は(2)の検索を複数回実行する）。

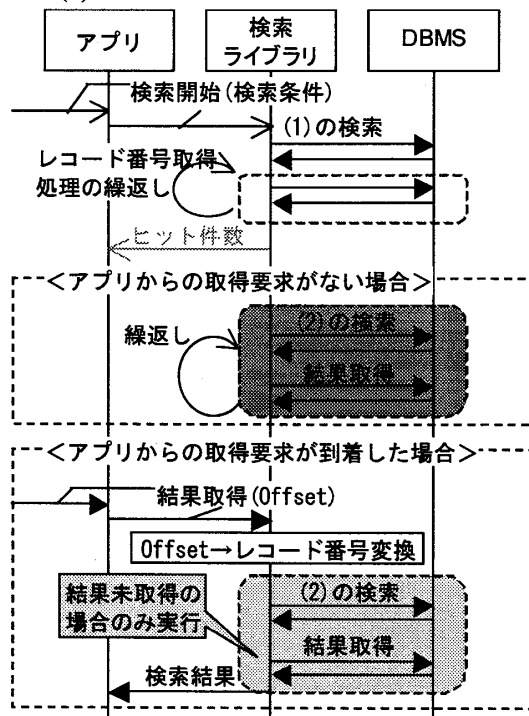


図2 提案手法概要

提案手法の概要を図2に示す。提案手法ではアプリケーション（図中では「アプリ」）と

DBMSの間に検索ライブラリを配置し、アプリケーションが検索要求を呼び出すと、検索ライブラリは検索要求を2種類の検索要求に変換した後、(1)の検索を実行し、レコード番号のみをDBMSから取得する。このとき得られるレコード番号の一覧はアプリケーションからの検索条件を満たした並びになっている。レコード番号の取得が完了するとヒット件数（取得したレコード番号の数）をアプリケーションに通知する。この時点でアプリケーションは任意の位置の検索結果を取得可能である。

アプリケーションから検索結果取得要求が発行されない場合、検索手法では(2)の検索を繰返し実行して検索結果をメモリ上に蓄積すると同時に、アプリケーションからの要求を監視する。アプリケーションが検索結果取得要求を発行すると、繰返し動作を一時的に停止させ、アプリケーションから指定されたOffsetに従い(2)の検索を実行し、検索結果をアプリケーションに渡す。このとき、既に検索結果をDBMSから取得済の場合はDBMSにアクセスせずに取得済の検索結果を渡す。

### 4. 考察

提案手法を用いることで、2章で述べた問題を解決できる。まず、アプリケーションからは検索開始要求を1回発行するだけで、ヒット件数が得られ、その後検索結果も取得できることから、同じ条件の検索を繰り返す場合より応答時間を短縮できると考える。次に、検索結果取得要求がアプリケーションから発行されたとき、提案手法では先にレコード番号の一覧を取得しているため、任意のレコードをDBMSから取得できる。したがって、検索結果全体が揃っていない状態でも「10ページ先」または「最終ページ」といったページ移動が可能となり、結果としてユーザ操作に対する応答時間の短縮が可能となる。しかし、提案手法には2種類の検索要求をDBMSに発行することによるオーバーヘッドの増加というデメリットもある。

### 5. まとめ

本稿では、組込み向けデータベースにおいてアプリケーションからの検索要求を2種類の検索要求に変換することで、ヒット件数をアプリケーションへ通知するとともに、任意の位置にある検索結果の取得を同じコストで実行できる検索方式を提案した。

今後は、試作を行い、提案手法の有効性を評価する予定である。