

オブジェクト指向データベースの性能評価と問題点 (2)

1E-2

佐伯 剛幸 安村 義孝 木村 裕

NEC C&C 研究所

1 はじめに

データベースベンチマークは、データベースの基本的な性能を評価し、データベースの構造的な問題点を浮き彫りにすることができる。現在、エンジニアリングデータベースの性能評価法として、OO1[Cattell 92]、ハイパーモデルベンチマーク、OO7[Carey 93]などが提案されている。本論文では、これらのうちのOO7を用いて、2つの市販のオブジェクト指向データベース(OODB)の性能評価を行なった結果とそれに対する考察を述べる。

2 OO7ベンチマーク

OO7ベンチマークのデータ構造を図1に示す。図の四角形とその傍の名前は、それぞれ、オブジェクト(インスタンス)とクラス名を表す。また、右側の数字は、オブジェクトの数を表す。

Module オブジェクトの下には、Assembly オブジェクトの階層がある。Assembly 階層は、Complex Assembly オブジェクトと Base Assembly オブジェクトから構成される。CompositePart オブジェクトは、Base Assembly オブジェクトと M:N の関係を持っている。1つの CompositePart オブジェクトは、Small データベースで20、Medium データベースで200個の AtomicPart オブジェクトと関連を持つ。AtomicPart オブジェクトは、同じ CompositePart オブジェクト内で、他のいくつか(3, 6, 9)の AtomicPart オブジェクトと Connection オブジェクトを用いて関連付けされている(この数を fanout と呼ぶ)。

この構造のデータベースに対する操作としては、トラバース、問合せ(ルックアップ)、構造変更が用意されている。キャッシュが空の状態での測定値を Cold、その後の3回の測定値の平均を Hot とする。

3 測定

2つの市販 OODB について測定を行なった。キャッシュサイズは、すべて、12M バイトに設定し、また、データベースをローカルとして使用した。なお、ディスクアクセス時間は、OODB2のマシンのディスクの方が高速であった。

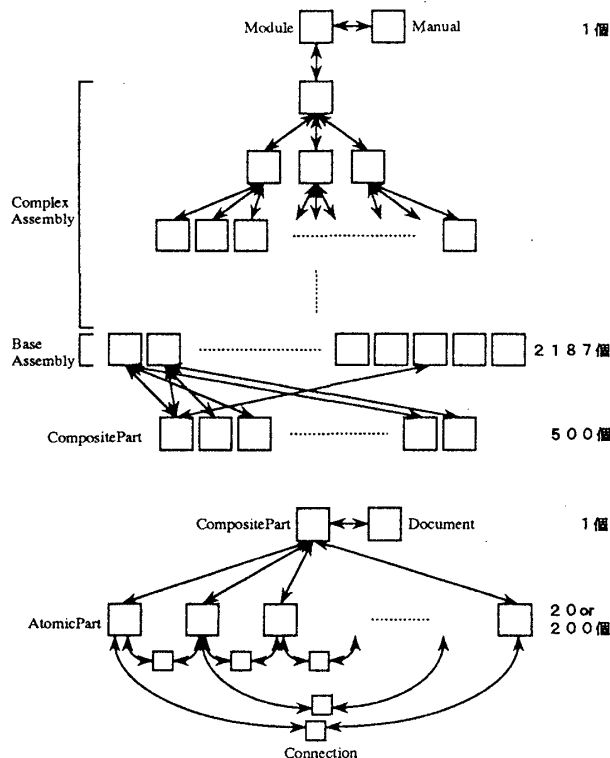


図1: OO7ベンチマークのデータ構造

OODB1

NEC EWS4800/220 (R3000, 30MHz), 32M バイトメモリ

OODB2

Sun SPARCstation 330 (SPARC, 25MHz), 48M バイトメモリ

4 結果

全体的な傾向を述べると、Medium データベースにおける問合せの半数以上と構造変更について、OODB1の方が高速であった。それ以外の問合せとトラバースのほとんどについては、OODB2の方が高速であった。ここでは、特に問題が見られた操作について、その測定結果を示す。

ワーキングセットサイズ(WSS)

問合せ Q7 は、「すべての AtomicPart オブジェクトをスキャンする」操作である。この操作での、Medium9 データベース (Medium データベースで fanout が9)での測定結果(経過時間、秒)を表1に示す。この結果を見ると、Hot の速度に向上がみられない。しかし、実際にアクセスする部分は AtomicPart オブジェクトのみ

Performance Evaluation for Object-Oriented Databases(2)
Takayuki SAEKI Yoshitaka YASUMURA Yutaka KIMURA
C&C Research Laboratories, NEC Corporation

表 1: Q7, Medium9

	Cold	Hot
OODB1	714.26	833.59
OODB2	2618.48	2506.11

であり、WSS はキャッシュよりずっと小さいはずである。

fanout の影響

問合せ Q8 は、「すべての Document オブジェクトと AtomicPart オブジェクトのペアについて、Document オブジェクトの id と AtomicPart オブジェクトの docId が一致するものを見つけ、一致したペアの数を返す」という操作である。この操作の、Medium、Cold での測定結果を表 2 に示す。この操作は、本来、fanout の影響を受けないはずであるが、OODB2 では、fanout が増加するにつれ速度が急速に悪化する。

表 2: Q8, Medium, Cold

	Medium3	Medium6	Medium9
OODB1	728.17	809.98	803.92
OODB2	3376.93	5457.53	6716.49

5 考察

Q7 については、実際の WSS を各オブジェクトのサイズから計算して推定したが、それは、ほぼキャッシュにおさまる量であった。Q8 については、fanout が変化しても実際にアクセスしているオブジェクトの総数に変化はなかったが、OODB2 では fanout が増加するにつれてディスクの入出力の回数が増加していた。このようなことから、以下のような推定を行なった。

CompositePart 以下の部分のオブジェクトの生成では、1 つの CompositePart オブジェクトに対して、それと関連するすべての AtomicPart オブジェクトを生成し、そのあと、それらに関連するすべての Connection オブジェクトを生成するということを繰り返している。だが、実際のディスク上での配置は、AtomicPart オブジェクトの間にかかなりの Connection オブジェクトが入っている。

このため、Q7 のような AtomicPart オブジェクトだけを持ってくる処理では、推定値よりもかなり多くのページをメモリに持つことになり、キャッシュのサイズを越えてしまうこととなる。また、Q8 では、fanout が増加すると Connection オブジェクトの数が

増えて 1 つのページに入る AtomicPart オブジェクトの数が減り、ディスク読み出しが増え速度が遅くなる。ただ、このようなルックアップ操作における速度の低下の度合はクラスタリングの方法に依存し、AtomicPart オブジェクトの間にそれと関連するほとんどの Connection オブジェクトを置くような方法を取っていれば、速度の低下は著しいものになると予想できる。このようなことから、OODB2 では OODB1 に比べて、複合オブジェクトを意識したクラスタリングをしていると推定される。

上の推定を正当化するいくつかの事実のうち 2 つを以下に示す。

1. CompositePart 以下の部分のオブジェクトについて、Connection オブジェクトを別のクラスタに格納するようにして測定したところ、Q7、Q8 についてかなり速度の向上が、T1 (Module から Assembly 階層、CompositePart、すべての AtomicPart をたどる) での速度の低下が見られた。これから、同じクラスタに格納すると、生成の仕にかかわらず複合オブジェクトを意識したクラスタリングが行なわれているとみなせる。
2. トラバースでは、複合オブジェクトを考慮したクラスタリングをしている方が高速になる。OODB2 は、トラバースでは全般に OODB1 より高速である。これから、OODB2 の方が複合オブジェクトを意識したクラスタリングをしているとみなせる。

6 まとめ

データベースの性能は、クラスタリングに大きく依存する。特に、クラスタリングがそのアプリケーションで頻繁に行なわれる操作に適合したものでなければ、Cold だけでなく、Hot での速度にも多大な影響を与える。これは、特に、大規模データベースにおいて顕著である。このようなことから、OODB においても、ユーザがクラスタリング方法を設定できるようにすることが必要である。

参考文献

- [Carey 93] M. Carey, D. DeWitt, J. Naughton. The OO7 Benchmark. *Proc. ACM SIGMOD Conf.*, June 1993.
- [Cattell 92] R. Cattell and J. Skeen. Object operations benchmark. *ACM Transactions on Database Systems*, 17(1), March 1992.