DBSJ Japanese Journal, Vol.13-J, No.1
October 2014

TPC-H ベンチマークの 100TB クラスを用いた商用アウトオブオーダ型データベースエンジンの評価と同クラスへの世界初登録

An Evaluation of Out-of-Order Database Engine by TPC-H Benchmark at 100TB Class and the World's First Registration to the Class

藤原 真二♥　茂木 和彦♥
田中 美智子♥　田中 剛♥
合田 和生♠　喜連川 優♣

Shinji FUJIWARA　Kazuhiko MOGI
Michiko TANAKA　Tsuyoshi TANAKA
Kazuo GODA　Masaru KITSUREGAWA

多機能情報端末の普及などにより，企業や社会活動で発生するデータが増加しており，ビッグデータ利活用への期待が高まっている．このような中，我々は内閣府最先端研究開発支援プログラムにおいて，アウトオブオーダ型データベースエンジン(OoODE)と称する実行原理に基づく超高速データベースエンジンの研究開発を推進した．本研究開発の成果を基に日立が製品化した商用アウトオブオーダ型データベースエンジンの大規模環境における有効性を確認するため，業界標準の TPC-H ベンチマークにおける最大のデータベース規模である 100TB のクラスを用いて性能評価を実施し，同クラスの性能測定結果リストに世界で初めて登録された．本論文では，この TPC-H ベンチマークの評価結果について報告する．

By the spread of smart devices and sensors, the amount and variety of data which is generated at the enterprise activity is increasing. Out-of-Order Database Engine (OoODE) was developed to utilize these big data efficiently. This research project was supported by the Japanese Cabinet Office's FIRST Program (Funding Program for World-Leading Innovative R&D on Science and Technology). OoODE is a high-performance database engine based on novel execution mechanism. Hitachi developed a commercial OoODE product based on the result of the OoODE research activity. This paper presents the performance evaluation result of OoODE product by using TPC-H benchmark at 100TB class.

♥ 正会員　(株)日立製作所
{shinji.fujiwara.yc, kazuhiko.mogi.uv, michiko.tanaka.ry, tsuyoshi.tanaka.vz }@hitachi.com
♠ 正会員　東京大学生産技術研究所
kgoda@tkl.iis.u-tokyo.ac.jp
♣ 正会員　国立情報学研究所，東京大学生産技術研究所
kitsure@tkl.iis.u-tokyo.ac.jp

1. はじめに

近年，クラウドコンピューティングの拡大や，多機能情報端末の急速な普及などを背景として，企業や社会活動で発生するデータが増加している．グローバルでの事業拡大や，新事業の創出，より豊かでスマートな社会の実現に向けて，ビッグデータ利活用に対する期待が急速に高まっており，データの超高速な検索処理を可能にするデータベース製品が求められている．

我々は，内閣府最先端研究開発支援プログラムにおいて，アウトオブオーダ型データベースエンジン(OoODE)と称する実行原理に基づく超高速データベースエンジンの研究開発を進めている[1]．本研究開発の成果を基にして，日立は2012年6月に商用アウトオブオーダ型データベースエンジン Hitachi Advanced Data Binderを製品化した[2]．

今回，同データベース製品の大規模なデータベース環境における有効性を，公的な基準の下で確認するため，意思決定支援の業界標準ベンチマークであるTPC-Hベンチマークの最大データベース規模(100TBクラス)を用いて性能の検証を実施した．この結果，同データベース製品が82,678.0QphH@100TBという優れた性能を達成したことが認定され，同クラスにおいて世界で初めて登録された[3]．TPC-Hベンチマークでは，従来，30TBまでのクラスにのみベンチマーク評価結果が登録されており，100TBのクラスは未到達の領域であった．これにより，大規模なデータベースの検索処理において，我々が開発してきたアウトオブオーダ型の実行原理に基づく超高速データベースエンジンが優れた性能を発揮できることが，国際的な基準の下，公的に証明されたと言えよう．

本論文の構成は以下の通りである．2章ではTPC-Hベンチマークの概要を説明し，3章では商用アウトオブオーダ型データベースエンジンの実装方式について述べる．4章では100TBクラスを用いたTPC-Hベンチマーク評価結果を報告し，5章で本論文を纏める．

2. TPC-H ベンチマーク概要

TPC-H ベンチマークは，非営利団体の Transaction Processing Performance Council が定めるデータベースの業界標準ベンチマークのひとつであり，Decision Support 向けのベンチマークとして，1999 年にリリースされた．TPC-H ベンチマークはデータベースの規模でクラスが分かれており，2003 年に 30TB と 100TB のクラスが追加され，現在では，100GB～100TB の 7 つのクラスが定義されている．著者らは，当時，未登録であった 100TB クラスにトップレベルの性能で登録することを目標に掲げた．性能目標に関しては，当時の 1TB クラス以上に登録されていたトップ 10 スコアを参考に，80,000QphH@100TB に設定した[1]．

TPC-H ベンチマークは以下のテストで構成される．

(1) Query Validation テスト
TPC-H で規定される検索クエリ Q1～Q22 を，性能評価を実施するシステムと同一環境に構築した 1GB の DB(Qualification DB)上で実行し，結果を確認する．

[1] TPC-H，QphH は Transaction Processing Performance Council の商標です．

(2) ACID テスト
上記 Qualification DB に対して ACID テストを実施する．Atomicity テストは更新を伴う ACID トランザクションを用いて，コミットやロールバックの基本動作を確認する．Consistency テストは ACID トランザクションを多重で実行し，一貫性を確認する．Isolation テストは Read-Write 競合，Write-Write 競合，競合のない Read-Write トランザクションの Concurrent Progress など 6 種類のテストでトランザクション分離レベルを確認する．Durability テストは，ACID トランザクション多重実行中に電源断などのハード障害を発生させた上で，システムを再起動して，コミット済みトランザクションが回復できることを確認する．

(3) Load テスト
TPC-H ベンチマークで提供される DBGEN ツールで生成した 100TB のデータをロードし，Test DB を作成する．ロード後，必要に応じて統計情報を収集し，所定の Verification クエリを実行する．データロード開始から Verification クエリの完了までがロード時間として計測される．

(4) Performance テスト
上記 Test DB に対して，Power テストと Throughput テストを 2 回実施する．Power テストは，更新クエリ RF1/RF2 及び検索クエリ Q1～Q22 を規定の順序で 1 つずつ実行する．Throughput テストは，規定された多重度と順序で検索クエリと更新クエリを実行する．100TB のテストでは多重度は 11 以上と規定されている．

(5) After-Run Verification テスト
Test DB の整合性を確認するため，Performance テスト終了後に，After-Run Verification テストを実施する．さらにその後，Auditor によるリモート監査を実施する．

なお，Query Validation 及び ACID テストは TPC の公認 Auditor 立会いの下，実機確認を含めて監査を実施．Load 及び Performance テストは，結果ログによる監査と Auditor が指定した Verification クエリにて実施した．

3. 商用 OoODE 実装方式

OoODE は，図 1 に示すように問合せ処理をアンフォールドすることにより多数のプロセッサコアを活用する．さらに，複数の非同期 I/O を同時に発行することで標準的なディスクドライブが有する Native Command Queuing 機能を効率的に活用する．このように OoODE は，サーバとストレージの性能を最大限引き出すことで性能向上を図る．

アンフォールドされた処理であるタスクの数は数千以上となるため，タスク管理オーバヘッドの低減が課題である．商用 OoODE では，OS のスレッドスケジューリングや管理オーバヘッドの影響を低減するため，スレッド内に複数のタスクを動作させるタスク内複数スレッド方式を採用した[4]．本方式により，多数のランダム I/O を伴うネスティッドループ結合処理に関して，スレッド数に比例した性能を得ることができた．

一方，TPC-H ベンチマークでは，ハッシュ結合など，シーケンシャル I/O を伴う処理が検索クエリの 9 割以上を占める．商用 OoODE では，ハッシュ結合処理方式として，ハイブリッドハッシュ結合方式を実装した．ハイブリッドハッシュ結合は，ハッシュ表がメモリに収まる場合にバケット分割を省略する処理方式であり，Build 処理と Probe 処理で構成される[5]．商用 OoODE では，多表ハッシュ結合において，各々の Build 処理や Probe 処理をアウトオブオーダ型で実行する．本処理にも，前述のタスク内複数スレッド方式を適用することで，スレッド制御オーバヘッドの低減と，ハードウェア資源の効率的な活用を実現した．

トランザクション制御方式は，参照時にロック競合が発生しない Multi-version Concurrency Control (MVCC) プロトコルを実装した[6]．MVCC には，いくつか実装方式があるが，商用 OoODE では更新トランザクションを 2-phase Lock で，参照トランザクションを Multi-version Timestamp Ordering (MVTO)で実現する Read-only Multi-version プロトコルを採用した．更新処理は，旧バージョンをタイムスタンプで無効化し，新バージョンを追記することで実現した．

トランザクション分離レベルとしては，READ_COMMITTED 及び REPEATABLE_READ を実装した．REPEATABLE_READ では，トランザクション開始時にコミット済みのデータを参照するスナップショット分離方式を採用しているため，Phantom Read は発生しない．TPC-H の ACID テストでは，規定に従い，上記の分離レベルを適用した．

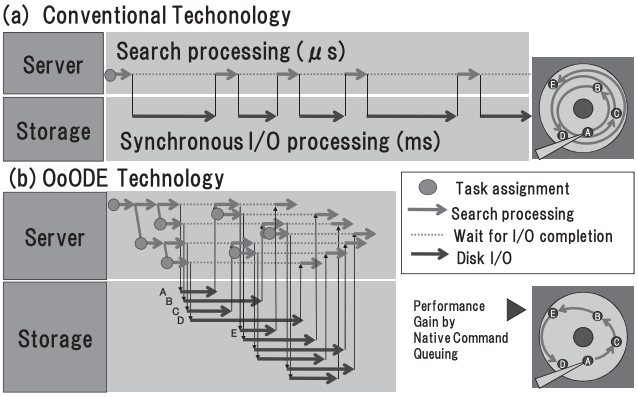

図 1　OoODE 実行原理

Fig.1　Execution Principal of OoODE

4. 100TB クラスを用いた TPC-H 評価

本章では TPC-H 評価環境及び評価結果について述べる．

4.1 評価環境

TPC-H ベンチマークにおける最大のデータベース規模である 100TB クラスの評価環境を図 2 に示す[2]．サーバは，10 物理コア CPU を 8 ソケット，主記憶を 2TB 搭載する Hitachi Blade Symphony BS2000 を 4 台で構成した．各サーバには 4 台の I/O 拡張筐体を接続し，合計 32 枚の Dual-port 8Gbps ファイバチャネルアダプタを搭載した．一方，DB を格納するストレージは，100 台の 900GB 10Krpm SAS ディスクを

[2] Intel Xeon は，米国およびその他の国における Intel Corporation の商標です． Linux は，Linus Torvalds 氏の日本およびその他の国における登録商標または商標です． Red Hat は，米国およびその他の国で Red Hat, Inc. の登録商標もしくは商標です．

搭載した Hitachi Unified Storage 150 を 16 台で構成した．各ストレージコントローラは 16 ポートの 8Gbps ファイバチャネルポートを有しており，サーバ側のファイバチャネルアダプタのポートと 1 対 1 で接続した．

OoODE はフラッシュストレージ環境でも高い性能を発揮するが[7]，今回はハードディスク環境で評価した．OoODE は，ストレージ性能を効率よく引き出すことが可能であるため，2014/2/21 時点における TPC-H ベンチマークの 10TB クラス以上に登録された Accepted Result 及び Historical Result (インメモリ DB を除く)と比較してデータベース規模あたり 1/4 以下である 1600 台のディスクで構成した．

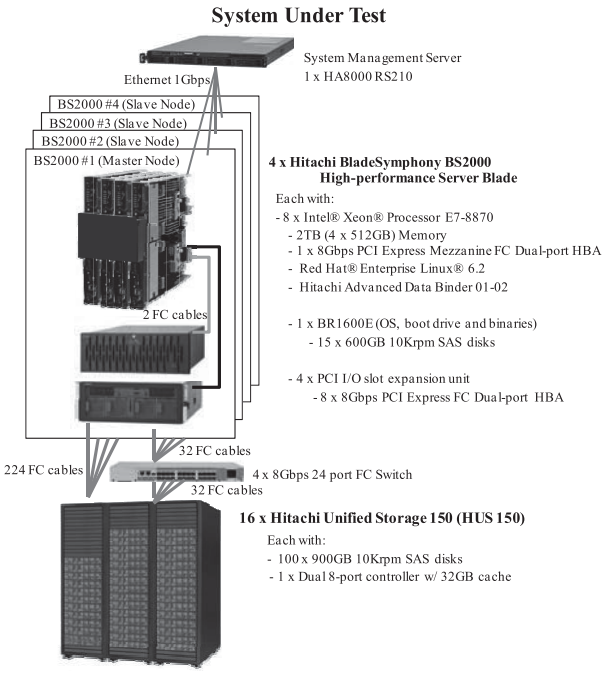

図 2 TPC-H ベンチマーク評価環境

Fig.2 Evaluation Environment of TPC-H Benchmark

4.2 Qualification DB の評価

Qualification DB は，性能テストの 1/100,000 の規模であるが，評価環境と同じハードウェア環境で実行する必要がある．そのため，100TB の構成同様，1GB のデータを OS が提供する LVM 機能を用いて 1600 台のディスクに分散格納した．その上で，Validation テストとして，規定された 22 個の検索クエリを実行し，結果が正しいことを確認した．

次に ACID テストを実施した．Atomicity，Consistency，Isolation の各テストでは，規定のシナリオに沿って複数のトランザクションを多重実行させ，データベースの動作を Auditor が確認した．Durability テストでは，Auditor の立会いの下，以下のテストを実機で実施した．

・サーバ障害：
　サーバ 1 台電源断
　全サーバ電源断
・ストレージ障害：
　HDD1 台抜き取り
　コントローラ片系障害
　コントローラ両系電源断

いずれも，ACID トランザクションを多重で実行中に障害を発生させた．そして，システム再起動後，DB を回復させ，コミット済みのトランザクションが回復されていることを確認した．

4.3 Test DB の評価

DBGEN で生成された 100TB の CSV 形式のデータを Test DB にロード後，性能テストとして，Power テストと Throughput テストを 2 回繰り返して実施した．Power テストは 1 台のサーバで，Throughput テストは 4 台のサーバで実行した．その後，After-Run Verification を実行し，最後に Auditor がシステムにログインしてリモート監査を実施した．最終的に，2 回目の測定結果である 82,678.0QphH@100TB が性能数値として認定され，2013/10/19 に登録された．TPC-H の 100TB クラスへの登録は，世界初である．

4.4 考察

今回評価した 22 個の検索クエリのうち，典型的な多表ハッシュ結合である Q8 と Q7 の評価結果についてベンチマーク実施後にサーバ 1 台で評価した結果を報告する．

評価に用いた Q8 とその実行プランを図 3 と図 4 に示す．Q8 は PART 表の選択率が小さいため，ハッシュ表がメモリに収まり，バケット分割処理は発生しない．

```
select
    o_year,
    sum(case
        when nation = 'SAUDI ARABIA'
        then volume
        else 0
    end) / sum(volume) as mkt_share
from
    (   select
            extract(year from o_orderdate) as o_year,
            l_extendedprice * (1 - l_discount) as volume,
            n2.n_name as nation
        from
            part, supplier, lineitem, orders, customer,
            nation n1, nation n2, region
        where
            p_partkey = l_partkey
            and s_suppkey = l_suppkey
            and l_orderkey = o_orderkey
            and o_custkey = c_custkey
            and c_nationkey = n1.n_nationkey
            and n1.n_regionkey = r_regionkey
            and r_name = 'MIDDLE EAST'
            and s_nationkey = n2.n_nationkey
            and o_orderdate between date '1995-01-01'
                                and date '1996-12-31'
            and p_type = 'SMALL PLATED COPPER'
    ) as all_nations
group by o_year
order by o_year;
```

図 3 Q8 クエリ

Fig.3 Query Q8

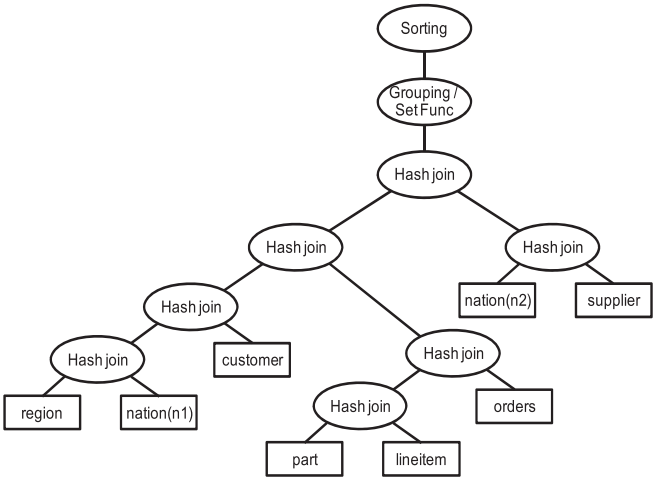

図 4 Q8 実行プラン

Fig.4 Execution Plan of Q8

図 5 は Q8 実行時の I/O スループットを示す. 実線は Read I/O，破線は Write I/O を示す．Q8 のハッシュ表はメモリに収まるため, バケット分割のための Write I/O は発生しない. フェーズ 1 は CUSTOMER 表と PART 表の検索結果をそれぞれハッシュ表として Build する．フェーズ 2 は，LINEITEM 表を Probe し，その結果をハッシュ表として Build する．フェーズ 3 は ORDERS 表を Probe し，その結果をハッシュ表として Build する．これ以外にも，SUPPLIER 表の Probe 処理などがあるが，それらは短時間で実行が終わっている. 図 5 で示すようにフェーズ 1 及びフェーズ 2 はいずれも 10GB とほぼサーバ 1 台の性能を使いきっている. フェーズ 3 は 2 つのハッシュ表の Probe 処理とその結果の Build 処理を同時に処理しているため CPU 利用率が 99%と CPU ネックになっている.

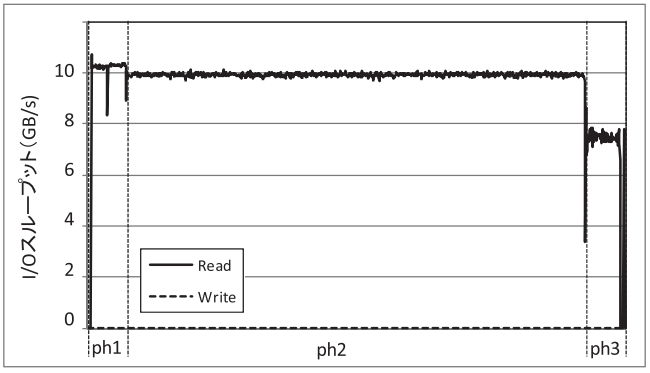

図 5 Q8 実行時の I/O スループット

Fig.5 I/O throughput of Q8

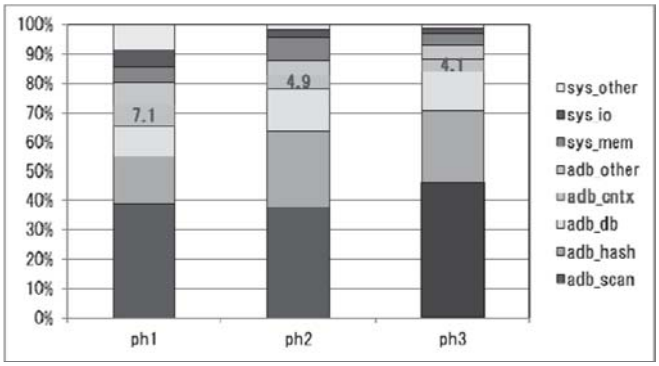

図 6 Q8 実行時の CPU 処理内訳

Fig.6 CPU processing detail of Q8

各フェーズにおける CPU 処理の内訳を図 6 に示す．本内訳では，CPU の idol 時間及び iowait 時間は除外している. sys は OS の処理，adb はデータベース処理を示す．商用 OoODE では，各フェーズの処理を数千のスレッドによりアウトオブオーダ型で実行している．しかしながら，DB 処理におけるスレッド制御(adb_cntx)の占める割合はフェーズ 1 で 7.1%，フェーズ 2 で 4.9%，フェーズ 3 で 4.1%であり，効率よく制御ができていることが判る.

次に，2 つめの評価に用いた Q7 とその実行プランを図 7 と図 8 に示す．Q7 は CUSTOMER 表及び LINEITEM 表がメモリに収まらず，バケット分割が発生する.

```
select
    supp_nation, cust_nation, l_year,
    sum(volume) as revenue
from
    (   select
            n1.n_name  as supp_nation,
            n2.n_name  as cust_nation,
            extract(year from l_shipdate) as l_year,
            l_extendedprice * (1 - l_discount) as volume
        from
            supplier, lineitem, orders, customer,
            nation n1, nation n2
        where
            s_suppkey = l_suppkey
            and o_orderkey = l_orderkey
            and c_custkey = o_custkey
            and s_nationkey = n1.n_nationkey
            and c_nationkey = n2.n_nationkey
            and ((n1.n_name = 'UNITED KINGDOM'
                  and n2.n_name = 'SAUDI ARABIA')
              or (n1.n_name = 'SAUDI ARABIA'
                  and n2.n_name = 'UNITED KINGDOM'))
            and l_shipdate between date '1995-01-01'
                               and date '1996-12-31'
    ) as shipping
group by supp_nation, cust_nation, l_year
order by supp_nation, cust_nation,  l_year;
```

図 7 Q7 クエリ

Fig.7 Query Q7

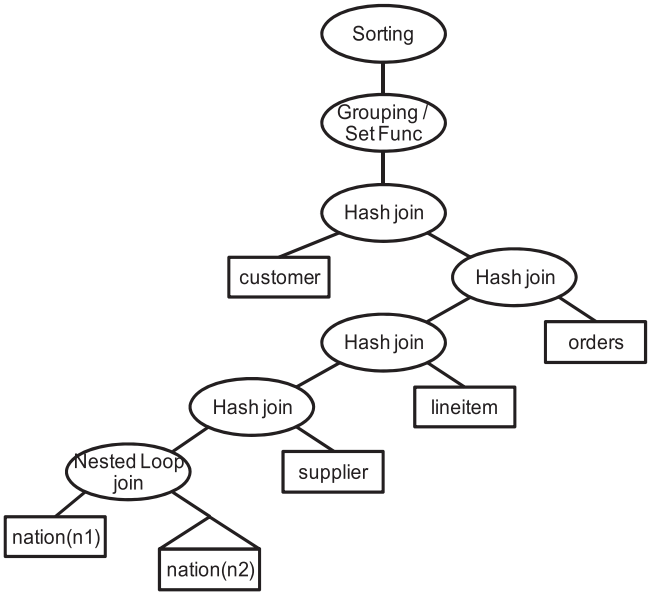

図 8 Q7 実行プラン

Fig.8　Execution Plan of Q7

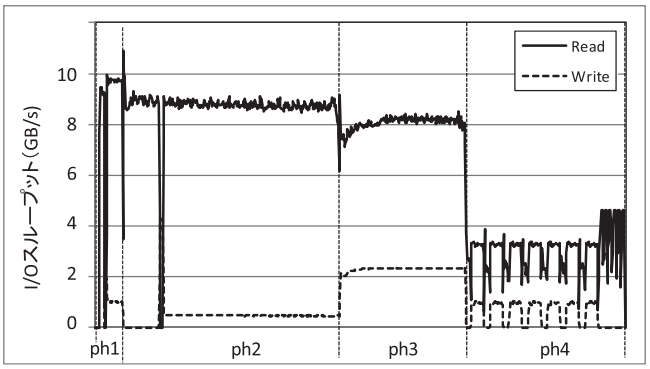

図 9　Q7 実行時の I/O スループット

Fig.9　I/O throughput of Q7

図 9 は Q7 実行時の I/O スループットを示す．フェーズ 1 では CUSTOMER 表の Build 処理を実行する．フェーズ 1 の途中から CUSTOMER 表のバケット分割に伴う 1GB/sec 程度の Write I/O が発生している．フェーズ 1 では SUPPLIER 表を Probe し，その検索結果をハッシュ表として Build する処理なども同時に実行されているが，短時間で終了している．フェーズ 2 は LINEITEM 表を Probe し，その結果をハッシュ表として Build する．フェーズ 2 の途中から Write I/O が発生しており，バケット分割を実行していることが判る．フェーズ 3 は Orders 表の Probe とバケット分割処理を実行する．フェーズ 4 はフェーズ 2 及び 3 で出力されたバケット毎のハッシュ結合を実行する．フェーズ 2 の結果がバケット分割されているため，処理の前半では，1 段目のハッシュ結合の中間結果のバケット分割を実行している．図 9 に示すとおりフェーズ 1，フェーズ 2，及び，フェーズ 3 は Read I/O と Write I/O の合計でサーバ 1 台の I/O 性能である 10GB をほぼ使い切っている．フェーズ 4 は 2 段のハッシュ結合と集計を同時に実行しているため，CPU 利用率が 93%と概ね CPU ネックとなっている．

各フェーズにおける CPU 処理の内訳を図 10 に示す．本内訳では，CPU の idol 時間及び iowait 時間は除外している．Q8 実行時と同様に，数千のスレッド制御のオーバヘッドは，数%以下である．

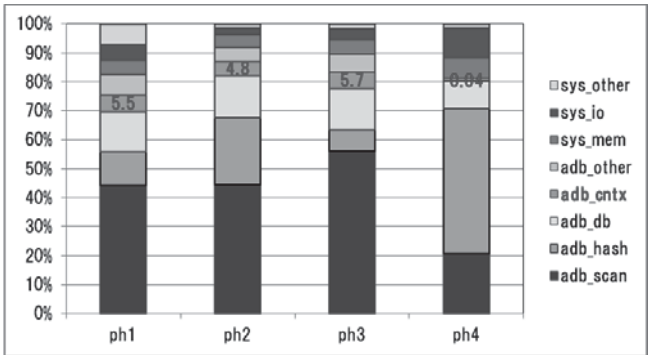

図 10　Q7 実行時の CPU 処理内訳

Fig.10　CPU processing detail of Q7

5. おわりに

本論文では，著者らが開発を進めている成果を利用した商用アウトオブオーダ型データベースエンジン Hitachi Advanced Data Binder の TPC-H ベンチマーク 100TB クラスへの世界初登録について報告した．同データベースエンジンは，トランザクション制御方式として，MVCC プロトコルを実装し，TPC-H ベンチマークが規定する ACID 特性を実現している．また，同データベースエンジンは，Build 処理や Probe 処理をアウトオブオーダ型で実行するハイブリッドハッシュ結合方式を実装している．本論文では，TPC-H ベンチマークのクエリを用いた性能評価により，同データベースエンジンがシステムの I/O 性能を十分に引き出していることと，数千のスレッドを数%のオーバヘッドで制御できていることを確認した．

TPC-H は基幹業務における意思決定支援システムをモデルとして設計されたベンチマークである．近年，期待が高まっているビッグデータの利活用では，多種多様なセンサーや多機能端末などの社会インフラ基盤から発生する膨大な時系列情報を扱う応用が拡大する．そのため，今後は，膨大な時系列情報の分析など社会インフラ基盤への応用に則した評価を進めていきたい．

[謝辞]

本研究は，　内閣府最先端研究開発支援プログラム「超巨大データベース時代に向けた最高速データベースエンジンの開発と当該エンジンを核とする戦略的社会サービスの実証・評価」の助成により行われた．

[文献]

[1] 喜連川優，合田和生,アウトオブオーダ型データベースエンジン OoODE の構想と初期実験，日本データベース学会論文誌, Vol.8, No.1, pp.131-136 (2009).

[2] 日立，東京大学との超高速データベースエンジンの共同研究開発成果を製品化,

http://www.hitachi.co.jp/New/cnews/month/2012/05/0528.html (2012).
[3] Transaction Processing Performance Council, TPC-H All Results – Sorted by Performance, http://www.tpc.org/tpch/results/tpch_results.asp.
[4] 清水晃, 徳田晴介, 田中美智子, 茂木和彦, 合田和生, 喜連川優, アウトオブオーダ型データベースエンジン OoODE におけるタスク管理機構の一実装方式の評価, 第 5 回データ工学と情報マネジメントに関するフォーラム (2013).
[5] D. J. DeWitt, R. H. Katz, F. Olken, L. D. Shapiro, M. Stonebraker, D. Wood, Implementation Techniques for Main Memory Database Systems, Proc. ACM SIGMOD Conference, pp. 1-8 (1984).
[6] Gerhard Weikum, Gottfried Vossen, Transactional Information Systems: Theory, Algorithms, and the Practice of Concurrency Control and Recovery, Morgan Kaufmann Publishers (2001).
[7] 早水悠登, 合田和生, 喜連川優, フラッシュストレージ環境におけるアウトオブオーダ型データベースエンジン OoODE の実験的クエリ処理性能評価, 第 6 回データ工学と情報マネジメントに関するフォーラム (2014).

藤原 真二 Shinji FUJIWARA
(株)日立製作所 IT プラットフォーム事業本部 DB 設計部主管技師. 1988 京都大学工学部情報工学科卒業. 1990 同大学院工学研究科情報工学専攻修士課程修了. 同年, (株)日立製作所中央研究所. データベースシステムの研究に従事. 本会, 情報処理学会, 電子情報通信学会, ACM, IEEE 各会員.

茂木 和彦 Kazuhiko MOGI
(株)日立製作所 横浜研究所主管研究員. 1992 東京大学工学部電気工学科卒業. 1997 同大学院工学系研究科情報工学専攻博士課程修了. 博士(工学). 1998 (株)日立製作所 システム開発研究所. データベースシステムの研究に従事. 本会, 情報処理学会各会員.

田中 美智子 Michiko TANAKA
(株)日立製作所 中央研究所 プラットフォームシステム研究部研究員. 2004 九州大学工学部電気情報工学科卒業. 2006 同システム情報科学府知能システム学専攻修士課程修了. 同年, (株)日立製作所中央研究所. データベースシステムの研究に従事. 本会, 情報処理学会各会員.

田中 剛 Tsuyoshi TANAKA
(株)日立製作所 中央研究所 プラットフォームシステム研究部主任研究員. 1993 東京工業大学工学部電気電子工学科卒業. 1995 同理工学研究科情報工学専攻修士課程修了. 同年, (株)日立製作所中央研究所. コンピュータシステムの性能評価, データベースシステムの研究に従事. 本会, 電子情報通信学会各会員.

合田 和生 Kazuo GODA
東京大学生産技術研究所特任准教授. 2000 東京大学工学部電気工学科卒業. 2002 同大学院工学系研究科修士課程修了. 2005 同大学院情報理工学系研究科博士課程単位取得満期退学. 同年, 博士 (情報理工学). データベースシステム, ストレージシステムの研究に従事. 本会, 情報処理学会, ACM, IEEE, USENIX 各会員.

喜連川 優 Masaru KITSUREGAWA
国立情報学研究所所長, 東京大学生産技術研究所教授. 1978 東京大学工学部電子工学科卒業. 1983 同大学院工学系研究科情報工学専攻博士課程修了. 工学博士. データベース工学の研究に従事. 本会理事, 情報処理学会会長, 電子情報通信学会, ACM, IEEE 各フェロー. 電子情報通信学会データ工学研究専門委員会委員長, ACM SIGMOD Japan Chapter Chair, VLDB Trustee, IEEE ICDE, PAKDD, WAIM などステアリング委員歴任.