大規模DWHを活用したDBMシステム（下）

前号で述べた大規模データベースマーケティング（DBM）システム構築のプロジェクトは、データウェアハウス（DWH）を含むデータベース（DB）が巨大で、オンライン（O/L）のレスポンスやバッチ（B/T）処理時間といった性能面での要求が非常に厳しかった。システム構築時の性能面の検証ポイントについて詳細に述べる。

今回のプロジェクトの概要

前号で指摘したように、システム要件上のポイントは、巨大な情報（データ量）の処理および情報を処理する際の高レスポンスの要請である。このことから以下のシステム構成を採ることとなった。

(1)　システム構成上の特徴

データ量が非常に大きくかつ将来的にさらに増加する可能性もあるため、拡張性を考慮し、超並列マシンであるRS6000/SP（IBM社製）を採用した。

DBとしては実績のあるOracle（日本オラクル社）を採用した。時期的にOracle 8（最新バージョン）の実績はまだ少なくOracle 7の方が安全という意見もあったが、大規模DBの場合、パーティション機能が必須であるため、Oracle 8を採用した。

ディスクはミラー構成とした。最終的には約９ギガバイト（GB）のディスクが640本という大規模なものとなった。

(2)　データ規模

顧客数は数千万件（顧客ごと、契約ごと、また月ごとのサマリー表なども持つため、数千万レコードのテーブルが10近くある）、売上情報は月間数千万件で最大19カ月保持。その他請求データ、マスター類を合わせて、テーブル数で約70、データ量は800GB～１テラバイト（TB）近くになる。インデックスやOracleの一時表領域、ログなどを合わせると、全体では約2.5TBの巨大なDBとなった。

(3)　性能要件

①O/Lレスポンス

ダイレクトメールを送付するために「A地域で○月から○月までの間にXXX円以上の買い物をした、Y歳～Z歳までの女性をリストアップする」といった処理が代表的なものになる。１日に500件程度の処理要求があり、おのおの10分～60分程度で完了させる。

②B/T処理時間

毎日ホストから、顧客属性の新規更新データや売上データが送られてDB側の更新処理を行う。また、月次で顧客や契約のステータスの再計算、DBフルバックアップといった処理を行う。日次は約10時間、月次は約２日間でB/Tを終了させなければならない。

O/Lのレスポンスを確保するために、サマリー処理的なことをB/Tでかなり行わざるを

えなかった。B/Tの数も日次でさえ1000を超え、ユーザーが使用していたバッチジョブスケジューラーが管理できるジョブ数の限界を超えてしまうといった事態まで発生した。

B/Tの負荷を減らせばO/Lレスポンスが悪化する、O/Lを重視すればB/Tが時間内に終わらない、というバランス取りに最後まで苦労することになる。

性能問題へのアプローチ

当初、性能問題をクリアするための関心事は以下の３つであった。

①ノード／ディスクはいくつ必要か

②データは物理的なディスク上にどのように配置すれば効率的か

③アプリケーション（アプリ）の開発時に、どのようにコーディングすれば効果的か

これらの問題の解答を得るために、図１に示すようなアプローチを採った（図１参照)。

そのキーワードは、まず性能を規制する要素ごとの原単価（計算の基礎となる性能値）の実測値を把握することである。次にこの各原単価を基に代表的アプリでのO/L、B/Tのレスポンスを机上で計算して概要を把握する。ここでも問題アプリ、システム構成のチューニングを行っていく。さらに実機に近い構成で本番に近いデータでの実機テストを行い、机上計算の是非を確認していく。このような３段階のアプローチである。

以下に各タスクのポイントを順に説明する。

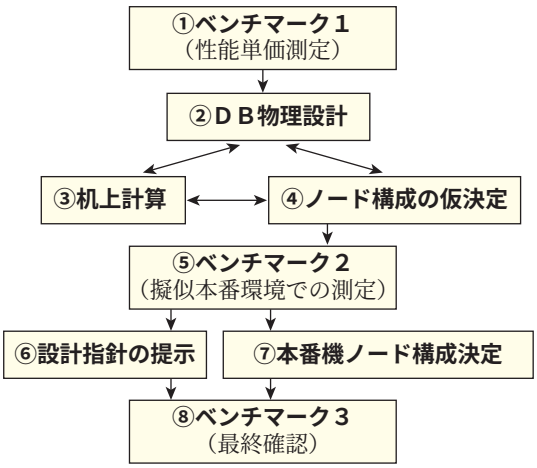

図１　性能問題解決のためのアプローチ手順

実機での基礎性能測定

①ベンチマーク１（性能単価測定）

処理時間を机上で計算する。DB物理設計を行うための元ネタとして、①ディスクI/O速度、②各種SQL（構造化照会言語）の速度(Select、Update、Insert、Loadingなどがおのおのxx件/秒)、③処理を多重化した場合の性能特性――などの項目について、実マシン上で性能測定する。

最適なDB物理設計

②DB物理設計

「①ベンチマーク１」で計測した結果を基にDBの最適な配置を設計する。

「①ベンチマーク１」で測定した「同一のチャネルから複数のディスクを同時にアクセスする場合」「他のノードからディスクをアクセスする場合」などの性能劣化を考慮して、アプリ処理特性から各テーブルを分割／配置

していく。
「このデータとこのデータは同時にアクセスされる場合が多いので別ディスクに置く」「このデータとこのデータは同一ノード配下に置かないと効率が悪い」といったことを考えながら設計していくのだが、ディスク本数320本、テーブル数70、パーティション総数2000といった規模となると担当者は「ジグソーパズル」と呼んでいた。Oracleに精通したメンバーおよびOracle社のコンサルタントが担当したが、非常に時間と根気を要する作業であった。

性能仮説の設定

③机上計算、および④ノード構成の仮決定

机上で各処理の所要時間を計算し、必要となるノード数などのシステム構成の仮説を導き出す。

現実的ではない所要時間が算出された場合（ノード数を増やす／Oracleチューニングなどでなんとかなるレベルではない）は、そもそもの処理方式を見直すことになる。

今回の処理では、数千万件のUpdate（更新）処理になってしまうものがあった。OracleでのUpdateはせいぜい250件/秒ぐらいなので、計算すると数十時間かかることになる。このため、他のマシン（ホストなど）でシーケンシャル処理を行いOracle側ではLoading＆Index作成のみ、といった変更をいくつかの処理で行った。

「うわっ、月次処理が１カ月以上かかってしまう」といった会話をしていたのがこのフェーズである。

性能仮説の検証

⑤ベンチマーク２（擬似本番環境での測定）

上記仮説に基づき、必要と思われるノード／ディスクを用意し、データも本番時に想定される量を用意して環境を構築する。

この環境の上で、擬似的なアプリを処理させて時間を計測する。

ある程度まとまった数のノードで処理させてみるのはこの時が初めてとなるため、特にクエリー（DBに対する処理要求）のパラレル度を変えながら、何ノードでどのくらいの性能、といったことを確認しながら計測し、最終的にノードやディスクの数などを決定する。

⑥設計指針の提示、および⑦本番機ノード構成決定

大規模なDBをハンドリングする場合、SQLの書き方で処理時間が数百から数千倍になる、といったことは日常茶飯事である。

したがって、「⑤ベンチマーク２」の結果から、「最適なコーディング方法」をアプリの開発部隊にフィードバックする必要がある。すなわち、「〜〜の場合は、ネステッドループジョイン（複数テーブル接続のアルゴリズム）より、ハッシュジョイン（複数テーブル接続のアルゴリズムでパラレル検索と組み合

わせ使用が可能）の方が速い」「〜〜テーブルを全地域分アクセスする場合はパラレル度はXX、地域指定でアクセスする場合はパラレル度YYをHINT文（プログラムの命令文）で指定しろ」といったことを整理して、開発部隊に提示する。

実際のプロジェクトでは、この部分がなかなかうまく伝わらず、後になって「聞いていない」「あれは、そういう意味だったの？」的なずれが多数発生してしまった。

「アプリを知っている技術屋」「技術を知っているアプリ屋」の必要性を痛感している。

⑧ベンチマーク３（最終確認）

総合テストのフェーズで最終的に性能を確認する。「⑤ベンチマーク２」の時とは大きく以下の２点で環境が異なるため、必ずしも期待どおりの結果が得られるわけではない。

●アプリの違い

詳細設計が進むにつれ、アプリの処理が変わっている場合がある（基本設計時の考慮漏れ、仕様変更など）。

●データの違い

「⑤ベンチマーク２」で本物のデータを使用できていれば問題ないが、使えなかった場合は自分たちでデータを作るしかない。件数はある程度同じにできるが、各項目のカーディナリティ（場合の数）やその分布は違ってくる。たとえば、「１カ月に100回以上カードを使う人なんていない、もしくは非常に少ない」と思っていたのが、実際には「数万人といったオーダーで実在する」といったことがあり、性能への影響が出る場合がある。

「⑧ベンチマーク３」で問題が発生するのは上記のような理由である場合が多いが、「①ベンチマーク１」や「⑤ベンチマーク２」の結果から再度最適になるようチューニングしていくことは十分可能である。

最後に

今回のシステム構築を踏まえると、以下のような条件を持つシステム構築は、必ず基本的かつ本格的な方式設計、性能評価が必要である。性能問題を単なる１タスクと過小評価するとプロジェクトの命取りになりかねない。

①大規模CSS構築
②大量の情報（データ）処理が必要
③要件として高い処理性能の要求
④新技術の採用（世の中の最先端技術、使用実績がないなど）

また、性能問題に対処するためには、顧客との交渉（仕様変更／運用／データ調達／マシン調達など）、開発者への指導（コーディングルールの徹底）、メーカーとの交渉（製品バグ対応／性能向上策の提示）など、全体をコントロールできる立場の者が積極的に取り組む必要がある。したがってプロジェクトマネージャーは「性能は基盤担当者のタスクの１つ」といったとらえ方をせずに、自ら積極的にこの問題に取り組むことが肝要である。

（野村総合研究所　水戸浩晶）