

データベースの動向と課題 日本ユニシスの取り組みと今後の動向

Trends and Challenges in database technology
Efforts and Challenges of Nihon Unisys

行 成 敦

要 約 データベース技術は、どのようなシステムでもバックヤードで利用される技術であるが、1990年代のシステムのオープン化の流れと共に、ベースとなるプロダクト及び利用方法は大きく変化した。関係データベースが一般化すると共に、メインフレーム時代のトランザクション中心処理から、データウェアハウスに代表されるように蓄積されるデータ量もメインフレームに比較して膨大な量になり、大量データの検索処理も重要な利用分野となっている。近年、オープンプラットフォームも商用 UNIX から Windows へ、そして Linux へと大きく変貌し、オープンソースで提供されるデータベース製品も登場し利用されてきている。

本稿では、まず、これまでのデータベースの歴史と日本ユニシスの取り組みについて振り返る。

次に、将来目指すべき方向性を検討した後、今後大きな流れとなると予想されるオープンソースの製品と既存商用製品を比較しながら、オープンソースにおいてシステムを実装する際の選択・検討の視点を提供する。

他方、データ量の飛躍的な伸長に伴い、今までの延長線上には見られない特化型データベース製品が出現している。その動向と日本ユニシスの取り組みについて紹介する。

Abstract Database technology has been used in any backyard system. In 1990s, the products underlying the technology and the usage of database systems have drastically changed. That is, as the relational database has become more popular, the volume of data involved has rapidly increased in the systems such as data warehouse compared with the traditional transaction centered system and the large volume data retrieval has become one of the important data management areas. Recently, the trend of computer platforms has moved from commercial UNIX to Windows and from Windows to Linux. With this moving, open source database products are becoming more popular.

In this paper, we look back over the history of database systems and our efforts in Unisys. After considering the future direction briefly, we discuss the difference of open source database systems and commercial ones from the viewpoint of the database features, and we propose the key points to implement successfully your systems using open source products.

On the other hand, a special purpose database system has emerged with the rapid increase of the data volume such as data streaming. We present our current effort for the new product.

1. はじめに

データベース技術は、どのようなシステムでもバックヤードで利用される技術であるが、1990年代におけるシステムのオープン化の流れと共に、ベースとなるプロダクト及び利用方法は大きく変化した。関係データベースが一般化すると共に、メインフレーム時代のトランザクション中心処理から、データウェアハウスに代表されるように蓄積されるデータ量もメインフレームに比較して膨大な量になり、大量データの検索処理も重要な利用分野となっている。

近年、オープンプラットフォームも商用 UNIX から Windows へ、そして Linux へと大きく変貌し、オープンソースで提供されるデータベース製品も登場し利用されてきている。

本稿では、まず 2 章で、データベースの歴史と日本ユニシスの取り組みについて振り返る。次に、3 章で将来目指すべき方向性を検討する。

4 章では、今後大きな潮流となると予想されるオープンソースの製品と既存商用製品を比較しながら、オープンソースにおいてシステムを実装する際の選択・検討の視点を提供する。

全く新たな方向性として、データ量の飛躍的な伸長に伴い、今までの延長線上には見られない特化型データベース製品が出現している。5 章で日本ユニシスの取り組みについて紹介する。

2. データベースシステムの歴史と日本ユニシスの取り組み 製品及び利用技術の視点から

本章では、これまでのデータベースの歴史を振り返り、その中で日本ユニシスが提供してきた製品及び取り組んできた技術・課題について紹介する。

2.1 関係データベースの普及 メインフレームからオープンシステムへ

データベースシステムは、1950 年代から見られ、ファイルベースの処理から、データを共有するシステムへと発展してきた^[1]。1960 年代には、データシステムズ言語協議会 (CODASYL) が発足、1970 年代にかけてデータベースのいくつかの概念を取り込んだ言語体系が整えられ、データベースの普及に大いに貢献した。CODASYL に基づくメインフレームの製品として、当時の UNIVAC^{*1} では DMS 1100、Burroughs^{*1} では DMS II を開発・リリースし、金融機関では、現在でもメインのデータベースとして利用されている。

一方、今日の主なデータベースの流れとして、1970 年 E.F. Codd により提唱された関係データベースの普及が挙げられる。「関係」という数学的概念をデータモデルに取り入れ、理論的根拠を与えると共に、非手続き的な操作によるデータ処理を提唱し、利用者からは非常に理解しやすい、利用しやすいシステムとして期待された。しかしながら、実装を意識しないモデルの提唱により、実装までは非常に長い道のりとなる。

1970 年代後半には IBM 社のプロトタイプ System R に始まり、Unisys 社^{*1} でも RDMS 1100 を開発し、1980 年代には関係データベースの提供・利用が始まった。ただ、まだ基幹系での利用には性能的に問題が多く、それまで銀行の勘定系などで利用されていた IBM 社 IMS 及び Unisys 社 DMS 1100 に置き換わるものにはならなかった。

関係データベースが普及する大きな契機となったのは、オープン系システムの台頭だと思われる。Sun Microsystems 社を初めとする UNIX 機の利用に伴い、UNIX 機上で稼働するデータベース製品が必要になるが、当時提供されるデータベース製品はすべて関係データベースであった。これは、上記性能問題が解消されたというよりも、「これから開発する製品」としては関係データベースしか選択がなかったと思われる。UNIX 機上では、Oracle 社、Sybase 社、Informix 社等複数のソフトウェア専門の会社から製品が提供されることになる。これに伴い、製品間の標準化を求める動きが始まり、ユーザインタフェースとして SQL が開発されることになる^{[2][3]}。SQL は現在最も利用されるプログラム言語の一つであり、データベースでは唯一の規格となっている。日本ユニシスでも、RDMS 1100 を開発・提供する必要性から、国際会議、国内 JIS 会議に積極的に参画し、標準化に貢献すると共に、早期に有用な機能を取り込み、いち早く市場に提供できるよう努めた。1980 年代後半には、RDMS 1100 を利用した基幹系の

経理システムも実装され成功を収めた。

ハードウェアの性能向上は、メインフレームに比較して、オープン系システムは目を見張るものがあり、性能を考慮した場合、関係データベースの利用は、オープン系システムへシフトしていくことになる。

一時期、Unisys 社でもメインフレーム上で稼働していたデータベース製品の UNIX への移植とプロトタイプ開発が進められていたが、この分野でのオープン系ソフトウェア製品の戦略は、市場に普及しているソフトウェアを利用することとなり、独自製品の開発は行っていない。

一方、新たな分野として 1990 年代初めにはデータウェアハウス用の機能を取り込んだ RDMS 1100 をベースとしたシステム DataCentral を提供、次いで Oracle データベースをパラレルに処理できるハードウェア Open Parallel Unisys Server(OPUS) を提供したが、市場への浸透は今ひとつであった。更に、ハードウェア的には UNIX 機から大規模 Windows システム ES 7000 へシフトしていく。

データベース製品の市場は、1990 年代後半には、Oracle、SQL Server、DB2 UDB の 3 プロダクトに集約されたが、日本ユニシスでは ES 7000 への注力と共に、SQL Server を中心としたシステム開発にシフトする。また、オープンソースソフトウェア(以下 OSS と呼ぶ)Linux の普及と共に、2004 年より SQL Server のみならず、ES 7000 上で Linux と Oracle を組み合わせたシステム構築を手がけている。実際、Unisys 社は 8 CPU を搭載した 6 台の ES 7000 上で Red Hat Linux 3.0 と Oracle Real Application Clusters のシステムを構築・提供している。

2.2 ミッションクリティカルシステムに対する取り組み メインフレームを中心として

関係データベースはオープン系システムを中心として急速に普及してきたが、金融勘定系など大量トランザクションが発生する、“真に”信頼性の求められるシステムへの適用は今日まで実現していない。利用方面として、社内システム、Web 連携など新たな分野及びシステムとして多様性の求められる情報系分野等に適用が限定されている。

Unisys 社では、金融機関の勘定系を典型例とする“真にミッションクリティカル”なシステムで求められる、信頼性の高いデータベース製品の提供をメインフレーム上で継続している。特に、1993 年に金融系パッケージ TRITON で採用された拡張トランザクションシステム(eXtended Transaction Capacity system: 以下 XTC と呼ぶ)は、先進的な技術を取り入れた画期的なシステムである^[4]。一つのデータベースを複数のマシンで共有するアーキテクチャを採用し、システム障害時には、トランザクションが更新中であったデータを除いて他のデータは全てのマシンから参照・更新可能で、他のトランザクションは影響なく処理が続行可能な非常に高い可用性を備えたものである。今日では、Oracle 社の提供する Real Application Clusters に類似したシステムであるが、ロックのメカニズムとして専用ハードウェアを利用するなど、ソフトウェアとハードウェアが一体となって、堅牢性、可用性、拡張性を引き出している。その後も、大量ユーザ、大規模メモリ対応を行い、当時のシステムは性能問題を引き起こすことなく今日まで利用されている。

2.3 先進データベースへの取り組み

旧来のデータベース CODASYL データベース及び関係データベースとは別の潮流として、1990 年代に登場したオブジェクト指向データベースがある。Oracle などの関係データベース

が普及する中、部品管理や CAD 等エンジニア系データ構造を関係モデルで表現する際に生じるモデルの意味的な限界及び実装時の性能問題が指摘され、実際にいくつかのベンダーから製品が提供され、パッケージに組み込まれた。

Unisys 社では、メインフレーム上で Semantic Information Manager (SIM) を提供し、営業支援システムを構築した実績を持っている^[5]。オープン系システムとしては、1990 年半ばには、オブジェクト指向データベース OSMOS を開発、米国では地図データベースの実装に利用された。OSMOS は、1995 年にオブジェクトワールドにおいて日本でも紹介され、多くの聴衆を集めた。しかし、オブジェクト指向データベースは、市場を拡大できず、多くの製品は撤退を余儀なくされている。その一番大きな理由は、適用分野として一般ユーザにとってはニーズがあまりなく、利用者が限定されていたことである。Oracle 等の関係データベースにもオブジェクト指向の機能は取り込まれているが、実際には利用者は少ない。また、ハードウェア性能の飛躍的向上によりオブジェクトモデルを直接利用しなくても、関係モデルの範囲内で複雑なデータ処理の性能問題がある程度解消されたことも大きい。データ構造に即した直接的なモデリングの問題は解決されていないが、今後、オブジェクトモデルのみを扱うデータベース製品が普及する見込みはないと思われる。

2.4 日本ユニシスの取り組み まとめ

データベース技術・市場の歴史と日本ユニシスの取り組みをまとめると表 1 となる。日本ユニシスの取り組みは、

- 1) DMS 1100 及び DMS II に代表されるミッションクリティカルなトランザクションシステムへの製品提供と適用、特に XTC に代表される先進技術を取り入れた信頼性を十分認識したシステム提供
- 2) UNIX, Windows 及び Linux 全てのプラットフォームでのシステム提供、Windows としては初めての大規模システム ES 7000 の提供と市場に広く利用されているデータベース製品の適用
- 3) オブジェクト指向に代表される、先進的な適用分野を目指した製品の提供及び適用において実績を挙げている。

現在、1) 及び 2) の経験を踏まえて、大規模ミッションクリティカルな金融システムの構築に取り組んでいる。今後も更に動向を見極めながら技術の選択、検証を行っていく方針である。

表 1 データベース分野の歴史

	1970 年代	1980 年代	1990 年代	2000 年代
技術動向	・ E.F.Codd による 関係データベース提唱	・ オープン系データベース 製品の出現	・ データウェアハウス出現	・ 大規模ミッションクリティカル分野への挑戦
業界動向	・ IBM 社 IMS リリース	・ IBM 社 DB2 リリース ・ Oracle リリース	・ Versant, ObjectStore ・ Partition 等大規模対応 ・ DWH エンジン: RedBrick	・ Oracle, DB2, SQL Server に 集約
日本ユニシスの取組み	・ DMS1100 リリース	・ RDMS1100 リリース	・ DataCentral, OPUS 提供 ・ SIM, OSMOS 開発・適用	・ ES7000 を中心とした 大規模ミッションクリティカル分野への適用

3. 新しい潮流と今後の挑戦

3.1 大きな二つの潮流

オープン系商用製品を利用したシステム構築に加えて、今後の新しいデータベースの方向性としては、大きく二つの潮流がある。

一つは、今までデータベースを利用するほどではない分野、例えばフラットファイルで業務が行われていた分野へのデータベースの利用である。これは、比較的少量のデータで、商用製品と同等のトランザクション機能やリカバリ機能が必要ではなく、プログラマ的には SQL によるデータ加工のしやすさが求められるものである。商用製品も安価であれば利用されてきたと思われるが、一般的にデータベース製品は高価なため、利用が見送られてきているという事情がある。このような分野では、Linux の普及と同じ理由で OSS の PostgreSQL や MySQL の利用が広まっている。但し、利用の可否については注意深く判断する必要がある。OSS 製品の利用に当たっては、4 章で詳細に議論する。

もう一つの方向性として、次世代データベースを目指した拡張機能が議論されている。現状、技術的に対応できていない分野への挑戦である^[6]。これについては、次節で議論する。

3.2 次世代データベースへの挑戦

関係データベースが普及し、データベースシステムとしての基本機能が実用レベルで提供されている状況を踏まえ、次世代データベースシステムの機能として次の 5 項目が挙げられている^{[7][8]}。

- 1) ハードウェアの革新的進歩を利用した、高速・大規模システムへの対応
- 2) 総費用を抑えるための自動管理機能の取り込み
- 3) マルチメディアデータ及び半構造データ等、現在殆どデータベースで管理されていないデータのデータベースへの統合管理
- 4) データアクセスパラダイムの変革（音声によるデータアクセスなど）
- 5) メタデータによるデータ管理を行い、統合された高速なアクセスの実現

いずれも、これまで部分的には実現されているが、更なる進化を遂げることにより、データベースの活用分野が広がることが期待できるものである。

1) は、近年のハードウェアの著しい進化に伴い、利点を最大限に引き出すための技術の必要性を示している。例えば、メモリ空間としては 64 ビットアドレスが一般的となり、一昔前のデータベースであれば全体が格納できるような大容量のメモリを搭載することが可能な環境では、データベースのメモリ構造及び操作としては、既存のアプローチとは異なる実装が必要である。単に、データをメモリに読み込むだけでなく、メモリ内での探索を高速化するアルゴリズムの開発・実装が必要となる。また、CPU 能力の飛躍的向上に伴い、今まで以上に最適化処理（オブティマイザ）に CPU を費やすことにより、全体としてデータ処理の向上を図ることも検討すべき項目である。一般的に 1) はデータベースシステム内部の問題であるが、データベース管理者もある程度の知識を備えて、特性を生かしたシステム構築を行うことが必要だと考えている。

2) は、ディスクの最適化処理、アクセスパスの最適化処理などデータベース管理者が行う仕事をシステムが自動的に行うことを目指すものである。具体的な機能の例として、最近では自動的に統計情報を採取して、自動的にアクセスパスを算出する製品も提供されている。この

機能は、データベース管理者の視点では非常に魅力的であるが、万能ではないことを理解し、管理業務を行うときに手助けしてくれる機能と位置付けて対応するべきである。

3) において、マルチメディアデータの取り扱いについては、1990 年代初めから提唱されており、SQL でもインタフェースが議論されている。バイナリ型のデータとしてデータベース内部に格納し、トランザクションで扱うことは既存の製品でも可能である。但し、検索はバイナリデータに付随した項目による検索に留まっているのが現状である。今後は、音声、画像処理と融合した形での「内容による高速検索」が求められる。また、半構造データの新たなデータ表現形式として、様々な分野で XML が大きな役割を果たしているが、データ管理の面からデータベースに格納して検索できる標準的インタフェースの早期提供が検討されている。

4) においては、音声、画像などによる全く新しい検索処理が模索されている。SQL は表形式のプリミティブなデータに対する問い合わせ及び結果取得を目的としたプログラム言語であり、音声や画像データを SQL 処理へ変換する技術が検討されている。

5) は古くからデータ辞書として様々な提案がなされてきた分野である。メタデータを統一的に扱うことにより、何がどこにあるか素早く見つけ、論理的に整合性のとれたデータを利用者に提供することを目的としている。メタデータを高速に検索して、その情報を元に本来のデータを探索するアプローチは、ハードウェアの高速化により、効率上の問題は解消されている。むしろ、どのような情報をメタデータとして管理すべきかという意味的な問題及び多くの人々が賛同する標準的なデータ定義の問題が残されている。データを標準化された形で検索または交換するためには、メタデータの標準化は必須であるが、これまでの試みは成功していない。今後成功するとすれば、強力なリーダーシップを持った政府・企業がリードする形で進められると思われる。

この5つのアプローチに対する実装方法としては、二つのアプローチがある。既存製品への拡張機能として取り組む方法と既存製品と全く異なる目的特化型の製品を提供する方法である。近年、特化型製品で注目を浴びているものが登場している。次節で詳細に説明する。

3.3 目的特化型データベース ストリーム処理エンジン (Stream Process Engine) など

前節で述べた新たな分野へのアプローチとして、Stonebraker は、汎用的なデータベース製品よりも特化型製品を開発・提供すべきであると提言している。その代表例として、時系列に格納される大量のデータを、リアルタイムに扱うことが必要とされるストリームデータ処理を挙げている。これは、金融のオンライン取引で発生するデータの情報をリアルタイムで検索する、電子タグから読み取った売り上げデータをオンラインで集計するなどの要求から生まれたものである。従来のデータベースを利用する限り、発生データを一度ディスク上のデータベースに格納して、それを検索する必要があり、ディスクの入出力のため、高速に大量データを処理して検索可能な状態にすることは難しい。ストリームデータを扱う専用データベースシステムでは、発生したデータをそのままインプットとして検索処理を行う。これにより、大量データのリアルタイムな情報が検索可能になる。

E.F. Codd が提唱した関係データベースに対する「12 の法則」及び OLAP (On Line Analytical Processing) システムに対する「12 の法則」と同様な位置付けで、ストリームデータ処理に求められる 8 個の機能が定義されている^[9]。特に、ストリームデータに特有な機能としては、以下の 3 つが挙げられる。

- 1) ルール1: データをディスクに格納する必要なく, ストリームデータをリアルタイムに処理できること.
- 2) ルール3: 不完全な状態に対して問い合わせ処理ができること.
ストリームデータ処理では, 問い合わせ条件に合致するデータが問い合わせ処理中に全てが揃うわけではなく, また到着順も想定したものとは限らない. このような状況に対応できること.
- 3) ルール4: 想定された状況を反映した結果を返すこと(時系列データを想定通り処理できること).

既存製品の拡張では対応できず, 専用のストリーム処理エンジンが有効に活用できる分野として以下のものが提示されている^{[10][11]}.

- 1) ストリームデータ
- 2) データウェアハウス
- 3) センサーネットワーク
- 4) テキスト探索
- 5) サイエнтиフィックデータベース
- 6) XML データベース

何れも次々と発生して格納される大量のデータを, リアルタイムに扱うことを目指しており, 米国で部分的に実績を挙げている. 日本でも, ユビキタス社会の実現に向けて実システムでの利用が待たれるところである.

このように, 専用データベースのアプローチも活発に模索されており, 日本ユニシスではデータウェアハウス専用マシンに取り組んでいる. 詳細は5章で紹介する.

データベースの潮流をまとめて図1に示す.

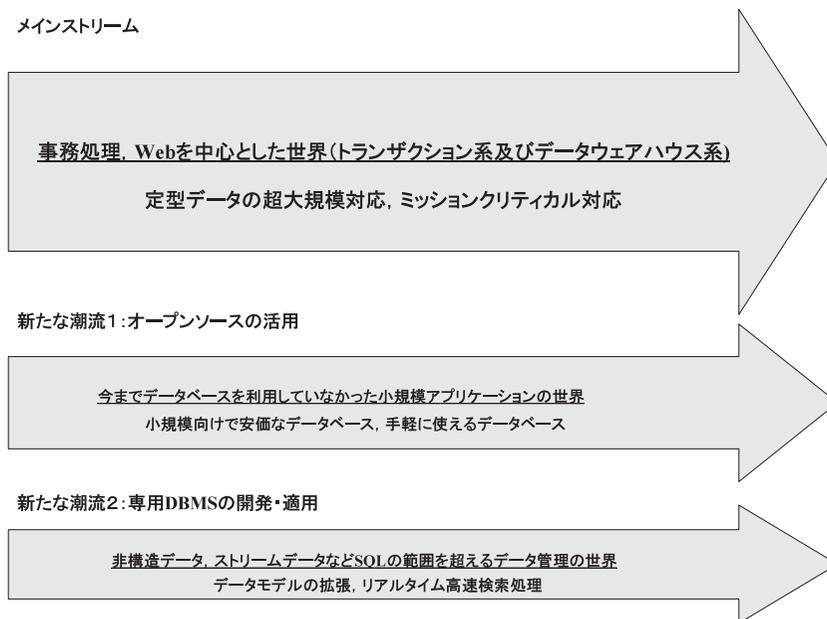


図1 データベースの潮流

4. 商用データベース製品と代替製品としてのオープンソースソフトウェア

4.1 選択のための機能面からの考察

データベース製品の一つの流れが、オープン系商用データベース製品の代替としてのオープンソースソフトウェアの利用である。商用製品のライセンスの高騰は一つの大きな要因ではあるが、サポートレベルの問題も同等以上に起因していると感じている。従来のメインフレーム時代は、製品自体に十分な障害解析の仕組みが備わり、OS、データベース等重要な基本ソフトウェアに対しては、殆どの障害原因の追究を行ってきた。一方、オープン系製品が様々なプラットフォームで利用され、圧倒的多数のユーザが利用する状況では、一つ一つの問題に対して原因を追究するよりも、当面問題を回避することに重点が置かれ、製品自体の品質については、メインフレーム製品と比べると容認される範囲が広い。このような状況では、もし製品自体の機能に差がなければ、商用製品とOSSとの違いはあまりないと言わざるを得ない。むしろOSSのほうが、ソースコードが開示されているだけ、能力さえあれば自力での解決も可能との認識も生まれる。従って、OSS製品の機能拡張によっては、商用製品が置き換えられる可能性は十分発生し得ると見ている。現実問題としては、以下に見るようにOSS製品の機能不足は明らかであり、採用に関しては十分検討・検証を行った上で利用する必要がある^{[12][13][14][15]}。

まず、機能の検討・比較にあたり、データベースシステムとして必要な要件を考察する。

データベース製品は近年OLTPのみならず、様々な分野に利用されており、技術革新とそれに伴う機能拡張は目を見張るものがある。しかし、ここではデータベースシステムを使う本来の目的に視点を戻し、製品群の機能比較、検討を行いたい。また、商用データベース製品としては、最も広く利用されているOracleをベースとして、適宜SQL Server及びDB2を参照する。OSSデータベース製品としては、広く注目を集めているPostgreSQL及びMySQLを対象とする。

データベースとして基本的に備えるべき要件は以下ものと考えている。

●実行時要件

- 1) トランザクション機能...ユーザ要求によるロールバック機能、Read Committedの独立性。
- 2) 同時実行制御...ロック制御としてレコード単位の粒度で行えること。
- 3) 性能...同時実行トランザクション数が増えた場合、想定範囲内で性能が保証されること。

●運用時要件

- 1) トランザクションリカバリ機能...ソフトウェア的な障害に対して、迅速にデータベース及びトランザクションが復旧できること。
- 2) メディアリカバリ機能...ディスク障害が発生した場合、指定した任意の時点の状態にデータベースを復旧できること。
- 3) バックアップ・リストア機能...バックアップ時間等運用可能な条件のもとで、バックアップの取得が可能なこと。また、想定した時間内に復旧できること。
- 4) データベースの再編成機能...再編成頻度が少ないまたは短時間でできること。

●開発時要件

- 1) 標準SQLインタフェース...少なくとも、基本的なSQLは提供されること。

2) 標準プログラムインタフェース...JDBC または ODBC が提供されること。

実行時要件 1) 及び 2) は、商用製品と OSS 製品では差がない。開発時要件については、基本的な機能について OSS 製品も提供している。実際には機能差はあり、提供している SQL の機能差は開発生産性の差に繋がるが、複雑なアプリケーションでなければあまり影響はないと考えている。

ここで注目したいのは、システム全体に影響を与える実行時要件 3) 性能及び運用時要件である。

性能は、1 台のマシンによるスケールアップを目指す場合、データベース製品の性能そのものが直接システムとしての性能に繋がる。従って、現状の OSS 製品は、スケールアップによる対応よりもレプリケーションによるスケールアウトが推奨されている。これは、大規模なシステムで且つ厳密なデータ整合性が求められる更新系システムには向かないことを意味する。採用検討にあたっては、注意が必要である。

一般的に性能は、アプリケーションに大きく依存し、実機での検証は必須である。これは、OSS に限ったことではなく、全ての製品に当てはまる。とはいえ、商用データベース製品はベンチマーク情報及び類似システムでの実績から、ハードウェアリソースの見積もりを行い、大まかな事前判断を下すことができる。一方、OSS 製品での事例は現在のところ、あまり重要でないエッジサーバ系での利用に留まっている。従って、現時点での採用を判断する場合、実機での十分な検証が必須となる。

運用については、製品によって大きく機能が異なる。Oracle のような商用製品は、どのような時点でもデータベースの状態を戻すことができる機能を提供している。また、バックアップ・リカバリについては、高速で行うための複数の選択肢を提供している。一方、OSS 製品ではフルバックアップを採取し、障害時にはバックアップ時点の状態に戻すのが基本的な発想になっている。このような方式は、以下のような問題点がある。

- 毎回フルバックアップを採取する必要があるため、業務を停止する時間が長時間となる。
- 局所的なファイル障害でも、全体を戻す必要があるため、リカバリ時間が長くなる。
- 復旧できるのは、バックアップ時点の状態であり、その後から障害発生時点までのトランザクション処理はアプリケーション等で復旧する必要がある。

以上のことから、OSS データベースは

- 更新頻度が特定され、更新後にバックアップが採取できるシステム

又は

- 復旧がバックアップ時点となることが許されるシステム（例えば、参照のみのシステム）のみに適用すべきである。フルバックアップを行う必要性から、データベースの規模の制約も受ける。最新版 PostgreSQL 8.0 では、Oracle と同様「ポイント・イン・タイム・リカバリ」の機能も提供されているが、リカバリは最後の砦になることから、十分検証を行った上での判断が必要である。

再編成については、PostgreSQL ではデータベースの構造上必須となる。PostgreSQL では削除又は更新されたレコードは、削除マークが付けられるだけで、全て残される。レコードは追加されるのみであり、再編成を行わないで更新を行うと、ファイルサイズが単調に増加する。これを回避するためには、再編成を定期的に行う必要があるが、業務時間の制約を受けることになる。

以上をまとめると、OSS 製品は、運用及び性能の視点から、参照中心の小規模データベースにまずは適用を検討すべきである。いずれにしても、運用及び性能については製品機能の十分な検討・検証を行った上での適用が必要となる。

4.2 選択のためのその他の要素

本節では、製品の機能性以外に製品を選択する場合の視点を議論する。

1) サポート体制の違い

メインフレームプロダクトとオープン系プロダクトの大きな違いは、最終的には、ユーザ数の違いに起因する。数の圧倒的な違いにより、オープン系プロダクトは膨大な技術情報及び事例情報が入手可能である。商用製品では、その情報は、ユーザフォーラムで提供されるものから、提供ベンダーが責任を持って提供するものまで多岐に亘る。商用製品が OSS 製品と異なる点は、提供ベンダーが公開している信用できる情報が存在すること、最終的な製品責任は提供ベンダーが持つことである。従って、OSS 製品を使用する場合は、自力である程度サポートを行っていく覚悟が必要である。或いは、強力なサポートを提供するシステム・インテグレータと共にシステム構築を行う必要がある。

2) 蓄積実績の違い

商用製品は、市場に提供されて 20 年近く経ち、様々なシステムに適用されてきた。その中には失敗した事例も多いが、利用技術の蓄積も膨大である。例えば、バックアップはどのように採取すべきか、チューニングのためには日常どのような情報を採取すればよいか、このような点についての報告及び事例は豊富に公開されている。一方、OSS 製品は、近年、ハードウェアのコモディティ化及び進化、並びに Linux の浸透と共に注目されてきたものであり、実績は小規模の限られた分野のみである。今後、新たな挑戦を行う企業が増え、製品自体の機能拡張も進み、既存商用製品の代替としての地位を確立する可能性はある。しかし、現時点では事例も少ないことから、十分な検証を行った上での採用が重要である。

3) 要員育成

データベースは、開発・構築と同等に運用フェーズのための技術の獲得が必須である。従って、本格的展開のためには、人材（開発・設計者と運用者、DBA）の育成をスコープに入れておく必要がある。これは、データベースに限ったことではないが、安定した運用が長期に亘って求められ、且つデータが日々増加・進化する環境を維持するためには、組織としての本格的な取り組みが必要となる。

4) コンティンジェンシプラン

OSS 製品を利用する場合、性能等の問題が発生した時のために、商用データベースに移行できるように工夫しておくことも必要である。

移行に当たっては、データの移行とアプリケーションの移行を視野に入れる必要がある。データ移行については日々のバックアップ等を用いて新たな商用製品による環境を検討する。アプリケーションについては、インタフェースは標準的な SQL をサポートしているため、独自機能を利用しないよう工夫することが大切である。使用プログラミング言語についても、JDBC などの製品でもサポートされているものを選択することが望ましい。

5) 利用者の心構え

サポート体制の項でも述べたが、OSS はあくまで開発者が自前で構築、運用していく姿勢が必要である。日本の場合は、システム・インテグレータがそれを担う場合が多いかもしれないが、ユーザサイドもプロダクトの選定背景を技術的に理解し、組織体制を整える必要がある。

自前で技術者さえ育てて保有すれば、長期に亘ってサポート可能なことが、OSS の利点の一つである。ソースコードが公開されているため、ベンダーに頼らず自力で問題解決できる可能性も秘めている。

以上、4章では、OSS 製品を利用していく上でのポイントを検討・整理した。日本ユニシスは、ソフトウェアベンダーに中立的な立場で、対象システムに適したデータベース製品を選択・適用できるよう取り組んでいる。

5. 大量データ検索処理対応としての特化型データベース

近年、生成されるデータは飛躍的に増大し、処理検索される対象も年々増加している。これは、既存のデータウェアハウスとして処理していたものに対して、何らかの対応が必要になっていることを意味する。対応として、UNIX など既存のハードウェア上に特化されたデータウェアハウスのソフトウェアを搭載することも提唱されたが、汎用的なソフトウェア製品を利用する方向に流れている。その要因としては、ソフトウェアのみ特化しただけでは、結局ソフトウェアに対する新たなスキル習得を管理者に求めるだけで、データロードに関する運用の負荷、効率チューニングなどの管理負荷は低減されないためである。管理者は、新たなソフトウェアのスキルと共に、それを既存ハードウェアと調和させて効率よく設計・運用するスキルが求められる。

この問題に対する新たなアプローチとして、大量データ検索処理に特化した、ハードウェアとソフトウェアを一体化したアプライアンスサーバに現在取り組んでいる。

本章では、日本ユニシスが現在取り組んでいる専用データウェアハウスサーバ Netezza について紹介する^{[16][17][18]}。

Netezza は、ハードウェア及びソフトウェア一体となったアプライアンスサーバである。アーキテクチャは、

- Teradata に代表される Massive Parallel Processing (以下 MPP と呼ぶ)
- DB 2 及び SQL Server に代表される Shared Nothing アーキテクチャ
- Oracle に代表される Shared Disk アーキテクチャ

の欠点を克服するためにハイブリッド構成を採用している。全体図を図 2 に示す。

高速化のポイントは、SMP Host で示されるフロントエンドと各データアクセスを司る Massive Parallel バックエンドに分離された構造 (Asymmetric Parallel Processing) による各パートの最適処理及びフロントエンド・バックエンド間でストリーミング技術を用いた高速クエリ処理 (Intelligent Query Streaming) である。これら先進技術を取り入れ、ソフトウェア、サーバ、ストレージを最適に組み合わせたアプライアンスサーバとして提供している。

データウェアハウスで必要とされる機能は、

- 1) 大量データに対する高速検索処理
- 2) 運用に耐えられる高速データロード

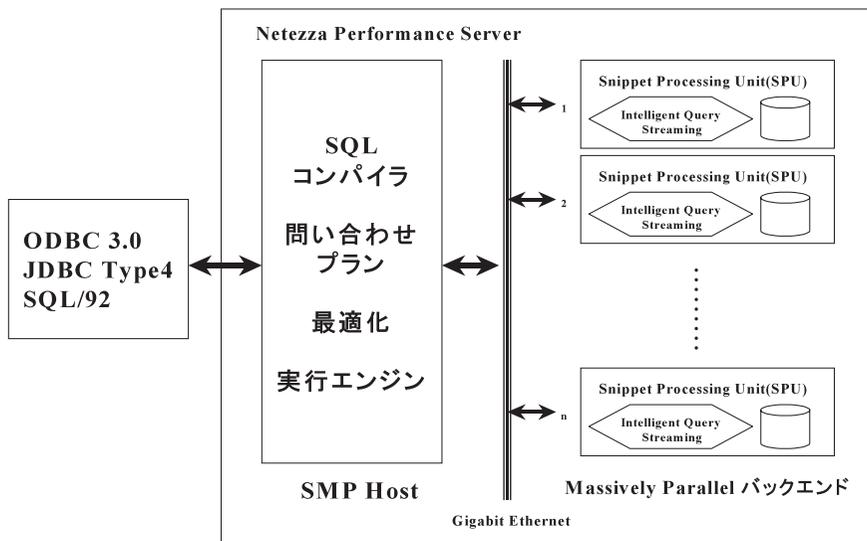


図2 Netezza Performance Server アーキテクチャ

3) クライアント分析ツール

である。

Netezza では、独自の高速ロードユーティリティを提供し、ODBC、JDBC の標準インタフェースをサポートすることにより広範囲なツールが利用可能である。

データベースマシンとしては、1990 年代データベース専用のハードウェアが商用化されたこともあったが、データベース製品の変更はその後の運用にも関わり、ユーザの運用を考慮するとスキルの獲得及び運用コストという点でデメリットが多く、受け入れられなかった。

しかし、データウェアハウスのような特定の処理をターゲットとすれば、ユーザに提供すべき機能は限定的であり、周辺のアプリケーションも少なく、特殊な実装を行っても、先進的な機能性の恩恵を受けられる。運用についても、ユーザサイドでは最低限の操作に対する知識を持てば十分である。日本ユニシスでは、Netezza を大規模データウェアハウスの一つのソリューションとして今後提供していく予定である。

6. おわりに

以上、これまでの日本ユニシスの取り組みを紹介しながら、データベースの歴史を振り返った。また、今後の潮流に従った製品の紹介及び日本ユニシスの取り組みについても紹介した。

オープンプラットフォームにおける大規模ミッションクリティカルなシステムへの挑戦は、現在取り組もうとしている最重要課題である。実績を踏まえて、今後順次紹介したい。

一方、1992 年の「次世代データベースの展望」^[19]では、オブジェクト指向技術やマルチメディア管理が今後の進む方向として示された。振り返ってみると、そのような「意味的」なデータ管理の問題解決はそれほど進んでおらず、ハードウェアの急激な進化に伴う、大規模化のみが大きく進化してきたように思われる。メタデータ管理、マルチメディア管理等応用分野での問題解決は、Web で利用される非構造データの管理を行う上では必須となる。今後の技術の進展に期待したいところである。

- * 1 1986年, 米国 Sperry 社と Burroughs 社が合併して Unisys 社が誕生した。日本では 1988年, 日本ユニバック(株)とパロース(株)が合併して日本ユニシス(株)が誕生した。UNIVAC は, 米国 Sperry 社のコンピュータ部門の事業ブランド名である。

- 参考文献**
- [1] 植村俊亮「データベースシステムの基礎」オーム社 1979年
 - [2] 芝野耕司「SQLの20年と現状および今後の展開(前編)」情報処理 Vol.45 No.5 May 2004
 - [3] 土田正士, 小寺孝, 芝野耕司「SQLの20年と現状および今後の展開(後編)」情報処理 Vol.45 No. 6 June 2004
 - [4] 阪口喜好「XTC UDSの概要」UNISYS 技報“特集: データベース技術”1992年5月
 - [5] 森 良行, 小野寺 裕「営業支援システムへの意味型データベース SIM の適用」UNISYS 技報“特集: データベース技術”1992年5月
 - [6] P. Bernstein, Stefano Ceri, M. Stonebraker, Jeff Ullman et al「The Lowell Database RESEARCH SELF ASSESSMENT」Communications of the ACM Vol.48, No.5 May 2005
 - [7] Pat Selinger「Top Five Data Challenges for the Next Decade」ICDE 2005 Presentation
 - [8] James Hamilton「interview : A Conversation with Pat Selinger」ACM QUEUE April 2005
 - [9] Michael Stonebraker, Uğur Çetintemel, Stan Zdonik「The 8 Requirements of Real Time Stream Processing」SIGMOD Record, Vol.34, No.4, Dec. 2005
 - [10] Michael Stonebraker, Uğur Çetintemel「“One Size Fits All”: An Idea Whose Time Has Come and Gone」ICDE 2005
 - [11] Michael Stonebraker「“One Size Fits All” An Idea Whose Time Has Come and Gone」ICDE 2005 Presentation
 - [12] 高橋 強「徹底比較!! Oracle & PostgreSQL」ThinkIT 連載 2004年11月~2005年12月
 - [13] 工藤 淳「商用&OSS データベースの現状と今後」ThinkIT 連載 2005年3月~2005年9月
 - [14] 「現場の選択: PostgreSQL vs MySQL」日経システム構築 2005年5月 no. 145
 - [15] 日本アイ・ピー・エム編著「IBMのLINUX戦略」IDG ジャパン 2003年
 - [16] 「Netezza Performance Server Appliance : An Architectural Comparison」Netezza Whitepaper 2005
 - [17] 「Business Intelligence in a Real Time World」Netezza Whitepaper 2004
 - [18] 「Tera Scale Data Warehouse Appliances Overcome the Technology Bottleneck」Netezza Whitepaper 2005
 - [19] 原 潔「次世代データベースの展望」UNISYS 技報“特集: データベース技術”1992年5月

執筆者紹介 行 成 敦 (Atsushi Yukinari)

1986年東京大学大学院理学系研究科修士課程終了, 同年日本ユニバック(株)入社。以来, データベース管理システムの開発及び保守に従事。メインフレームデータベース製品 UDS 1100の大規模機能開発, Unisys 製オブジェクト指向リポジトリ UREP 及びオブジェクト指向データベース OSMOS の適用支援を経て, オープン系データベース Oracle 及び日立製データベース HiRDB 適用支援に従事。現在はエンタープライズ IT サービス統括部プロジェクトサービス部に所属。