

なると列の数に比例したオーバーヘッドが見逃せなくなってくる。

HiRDBでは、列数が非常に多い表に対してでも、列数に依存しない安定した性能を維持できる行インタフェースを利用することができる。

HiRDBは1表中に定義できる列数は最大30,000であるため、基幹データベースからエクストラ・データウェアハウスに至る巨大な表に対しては、特に行インタフェース機能が性能を左右すると言える。行インタフェースを利用する場合は、定義系SQLのCREATE TABLE文にFIXオプションを指定するだけでよく、容易に導入することができる。

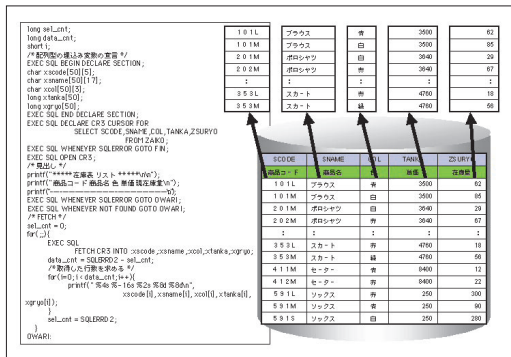
■ クライアント間の通信回数を削減する

ブロック転送機能

HiRDBクライアントからHiRDBサーバにアクセスし、大量のデータを検索する場合にブロック転送機能を使用すると、検索性能を向上できる。ブロック転送機能とは、HiRDBサーバからHiRDBクライアントにデータを転送するときに、任意の行数単位で転送する機能である。

サーバ側で1行ヒットするたびに、結果をクライアントに転送する方式では、ヒットした行数の回数だけクライアントーサーバ間で通信が発生することになる。しかし、ブロック転送機能を利用し、転送行数を多くすれば通信回数を削減することができ、トータルとしての検索時間（スループット）を短縮できるのが特長である。ブロック転送機能の設定方法も容易で、クライアント環境変数（PDBLK）に設定するだけでよく、UAPをリコンパイルしたり、新しいクライアントライブラリにリンクし直すという必要もない。

図4-5：配列フェッチ



■ 複数行の検索結果を取得する配列フェッチ

検索結果を取得するためにFETCH文のINTO句に埋め込み変数を指定し、検索結果を取得するのが一般的である。HiRDBでは、この埋め込み変数に配列型の変数を指定することが可能である。したがって、1回のFETCHで複数行の結果を取得することができる。これは、HiRDBクライアントからHiRDBシステムにアクセスし、大量のデータを検索する場合に非常に有効となる。

ブロック転送機能と異なるのは、複数行の検索結果を取得することをプログラム内で明示的に記述できるということだ。この両者の機能をうまく利用することにより、UAPの性能を大きく向上させることが可能である。

■ 正規形データを効率的に管理する繰返し列

繰返し列とは、1行中の1つの列に複数の値を格納できる可変要素数の配列である。これにより、非正規形のデータでも効率よく格納・管理することができる。繰返し列を使用すれば、従属するデータ数が定まらない場合でも、より自然な構造で扱え、表数を削減して検索時の結合を回避したり、行数を削減して行単位のオーバーヘッドを回避したりできる。以下に、その定義例と検索例を示す。

・ 定義例

```

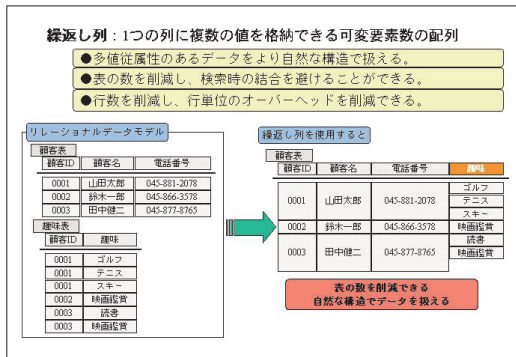
CREATE TABLE 顧客 (
顧客ID CHAR (10) ,
顧客名 NCHAR (10) ,
電話番号 CHAR (12) ,
趣味 NCHAR (30) ARRAY [10] )
    
```

・ 検索例

```

SELECT 顧客名 FROM 顧客
WHERE ARRAY (趣味) [ANY] ( 趣味 = 'テニス' )
    
```

図4-6：繰返し列



■ 段階的に対象レコードを絞り込む絞り込み検索

HiRDBでは、日立のメインフレームRDBMS [XDM/RD] でサポートしている絞り込み検索機能をサポートしている。絞り込み検索とは、段階的に対象レコードを絞り込む検索のことをいう。

絞り込み検索をする場合、操作系SQLのASSIGN LIST文でリストを作成する。リストとは、適当な件数になるまで条件を指定して段階的にデータを絞り込んでいくような情報検索をするために、その途中段階のデータの集合を一時的に名前（リスト名）を付けて保存したもの、または保存したデータの集合を意味する。作成したリストには、結果としてROWIDが含まれる。ある条件で作成したリストがあれば、そのリストを使用することで処理速度の向上が図れる。また、複数の条件を指定する場合は、複数のリストを組み合わせた検索も可能である。

絞り込み検索を使用するためには、リストを格納するためのリスト用RDエリアが必要となる。作成したリストが不要になった場合は、DROP LIST文で削除することができる。また、そのままにしておいた場合、次回、HiRDB起動時に自動的に削除される。

図4-7：絞り込み検索

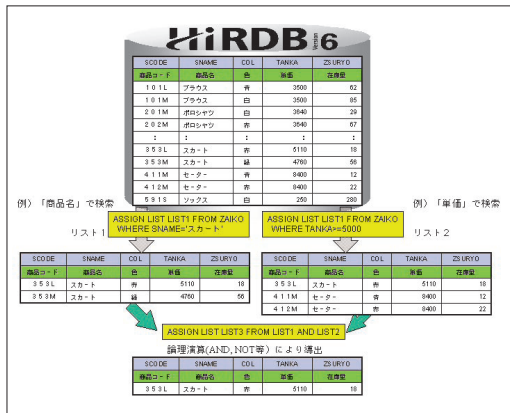
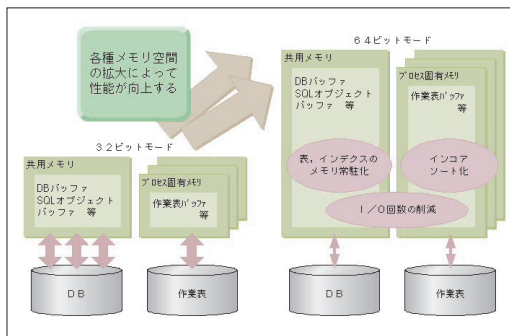


図4-8：64ビットモード対応



■ ネイティブ64ビットモード対応

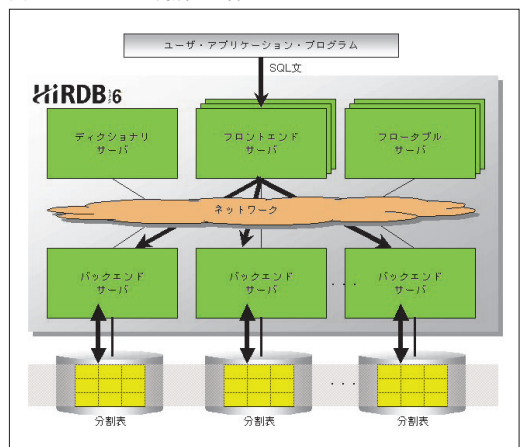
UNIX版のHiRDBは64ビットモードにネイティブに対応しており、大規模なデータベースシステムにおいて、各種バッファサイズを大きく確保できる。大きなバッファを利用することで、I/Oの回数を削減でき、性能向上が期待できる。

4.2 高性能とスケーラビリティを約束する並列アーキテクチャ

HiRDBにはSingle ServerとParallel Serverという2種類のタイプがあり、1台のマシンから複数台のマシンに至るまで、プロセッサ数、データベース規模に応じて選択することができる。Single Serverは、小規模から中規模なデータベース構築を目的としており、オンライン系トランザクションでのレスポンス性能を重視する場合に適したエンタープライズモデルである。Parallel Serverは、中規模から大規模なデータベース構築を目的としており、大量データ処理やデータロードの実行時間を重視する場合に適したエンタープライズモデルである。基本的にはデータベースの規模が選択の目安となり、1台のマシンに収まるような小～中規模データベースを構築する場合は「HiRDB Single Server」を、複数台のマシンで大規模なデータベースを構築する場合は「HiRDB Parallel Server」を、それぞれ導入することになる。

しかしながらその選択の際、HiRDBは将来のシステム規模の増大を心配する必要がなく、その後の業務拡大に伴ってデータベースの規模が大きくなった場合にも、サーバ台数を増やしてSingle ServerからParallel

図4-9：HiRDBを構成する各サーバ



Serverに拡張することができる。

HiRDB Parallel Serverは、ソフトウェア機能をいくつかの「サーバ」に分割して実装し、うまく負荷分散することで処理効率の良い並列DBMSを実現している。そのHiRDBを構成するサーバには、フロントエンドサーバ (FES)、バックエンドサーバ (BES)、ディクショナリサーバ (DS) がある。

まず、FESは受け取ったSQL文を解析して、必要なデータの存在するBESに処理を割り当てる。BESは各ノードに複数割り当てる。BESは入出力処理だけでなく、データの検出や並べ替えの処理も行う。そしてDSは、システム及び分割表などのデータベース構成情報を管理する。

1つの表がBESごとに分割して格納されているような場合は、FESで受け付けたSQLを複数のBESが並列に検索・更新することができるため、スケーラブルな性能を得ることができる。これが実現できるのもHiRDB Parallel Serverが、世界に先駆けてシェアド・ナッシング方式を採用した並列DBMSだからである。シェアド・ナッシング方式とは、複数のプロセサに対して個別のディスクとメモリを割り当てる方式であり、個々のプロセサは独立して動作できることを特長としている。

また、UAPからのSQLを受け付けるFESも複数設けることができるため、複数のSQLを同時に実行することも可能となる。このアーキテクチャにおいて、これらのサーバをうまく割り当てることにより、HiRDB Parallel Serverは負荷をうまく分散し、かつ並列処理を実現することができるのだ。

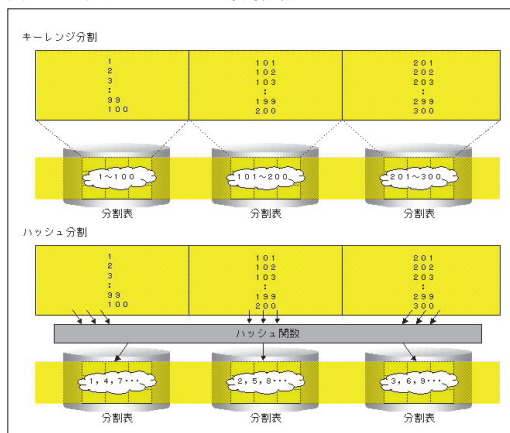
■ 負荷分散を実現する表やインデクスの分割格納

表を特定のRDエリアだけに格納すると、データ量が増加したり、アクセス頻度が高くなったりしたときに、そのRDエリアに対する負荷が集中しやすくなる。この結果、そのRDエリアに対するアクセス性能の劣化や、バックアップや再編成などの操作性の低下を招いてしまうことがある。

HiRDBでは、表のデータを複数のRDエリアに分割して格納することができるので (表の横分割機能)、表に対する負荷を分散させることができる。

また、バックアップや再編成などをRDエリア単位で操作することができる。さらに、パラレルサーバでは、表を複数のマシン (ノード) に分割して格納することができるので、並列処理によって表に対するアクセスを高

図4-10：表、インデクスの分割格納



速化することが可能である。

HiRDBの表の横分割方式にはキーレンジ分割、ハッシュ分割があり、表の用途に合った方式を選択することができる。キー値の範囲で分割格納先を決めるキーレンジ分割方式は、データ量を均等に分割させるなどDB設計時に考慮が必要となるが、データの格納先が明確であるため各レンジごとのデータメンテナンスが容易である。

一方のハッシュ分割方式は、キー値をもとにハッシュ関数を用いて分割格納先を決める方式であり、この場合、自動的に各格納先へほぼ均等な量のデータを割り付けることができ、DB設計の負荷を軽減する。さらに、ハッシュ分割方式に拡張性・柔軟性を持たせたフレキシブルハッシュ分割方式も用意している。

■ データ量の急激な変化にスムーズに対応するフレキシブルハッシュ機能

データ量が増加すると特定の表に負荷が集中することがある。こうした場合は、負荷が集中している表を分割

図4-11：フレキシブルハッシュ機能

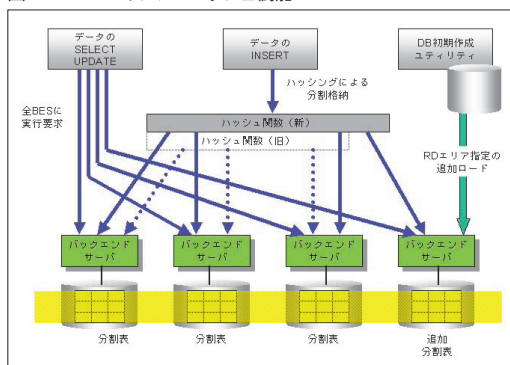


図4-12：並列バックアップ、並列リカバリ機能

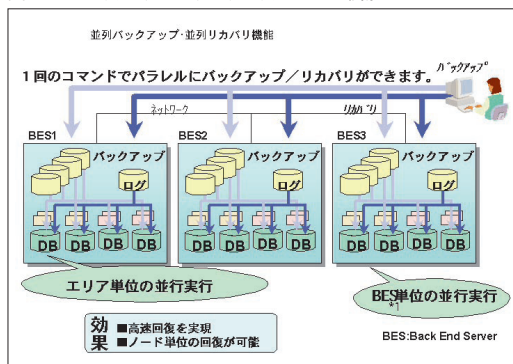
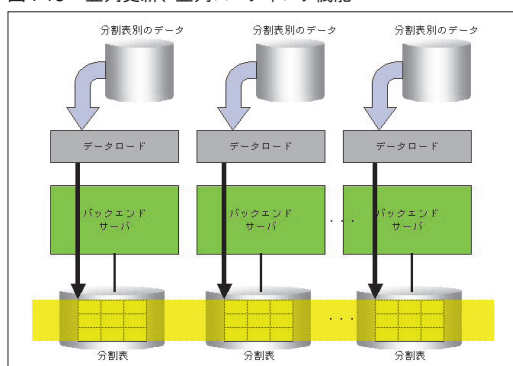


図4-13：並列更新、並列ローディング機能



することにより、ノード間の負荷バランスを再調整することになるが、通常はデータを再ロードして再配置するという作業が必要となる。HiRDBのフレキシブルハッシュ機能を使えば、ノードの追加と表の分割においてデータを再ロードするなどの手間を必要としない。ハッシュ関数は自動的に変更され、データは変更後のハッシュ関数に従って増設部分にも自動的に格納されるようになる。

■ 並列処理によるDB再編成、DBバックアップ、DB回復

Parallel Serverでは、DB再編成（ディスク内のフラグメンテーション解消）やDBのバックアップ取得、システム障害が発生した場合のDB回復などの処理も各サーバごとに実行できる。このため、DBの保守や障害時において、システム回復を短時間で行うことが可能になる。

■ 並列更新、並列ローディングにより大量データのバッチ処理時間を短縮

Parallel Serverでは、複数に分割した表に対し、検索

はもちろん更新なども並列に処理することで高性能を発揮。ソートやジョインなど負荷の重い処理は別プロセッサ（フロータブルなSQLサーバ）を動的に割り当てるとともに、パイプライン的に並列に処理することで検索結果出力までの時間を短縮する。

また、システム構築時などの大量データのローディングも、今までは10時間かかっていたものが10並列のデータローディングにより1時間ですませることも可能となる。これにより、データウェアハウスなどの大規模なシステムの運用が容易になる。

5 高品質なサポートサービスとコストパフォーマンスの高さ

5.1 | 自社開発ならではの優れたサポート

システムを安定稼働させ、日々の業務をスムーズに行うためには、サポートを含めた高品質なサービスが不可欠である。本誌の冒頭の「1.1 HiRDBの歩み」で述べたように、HiRDBは、1970年代から長年にわたりメインフレームで培ったデータベース技術を継承している。これは、長年にわたりデータベース製品開発を継続・発展させてきた結果であるが、製品や技術を実現している開発者の存在が大きい。今も開発経験豊富な技術者が製品開発を通じて若い技術者を育成し、先進技術を取り入れつつ進化しつづけている。この製品開発サイクルが優れたサポート力の源泉になっている。

HiRDB技術者のサポート力は、次のサービスの形でユーザに提供している。

- ・ 製品サポートサービス
- ・ DBシステム構築サービス

5.1.1 製品サポートサービス

高い信頼性を誇るHiRDBにも障害は起こりえる。万一トラブルが発生した場合でも、その影響を最小限に抑え、いち早くシステムを復旧するためには、製品の耐障害機能だけでなく迅速かつ的確な対応が可能なサポート体制が必要である。そんなニーズに応えるのが、製品サポートサービスである。製品サポートサービスは、基本サポートサービスとアドバンストオプション、ミッションクリティカルシステム対応サポートサービスから構成される。基本サポートサービスには、ソフトウェアの間

題解決支援や改良版の提供、バージョンアップが含まれる。アドバンストオプションは問題受付時間の夜間延長オプションである。これらサポートサービスを前提にして24時間365日の運用が必要な基幹業務システム向けにミッションクリティカルシステム対応サポートサービスがある。このサポートサービスでは、サポートチームを設置し重要障害の24時間対応やオンサイトでの問題解決支援、システムの定期診断による潜在的問題の早期発見と改善策の提案、といったサービスを提供する。

5.1.2 DBシステム構築サービス

DBシステム構築サービスは、信頼性の高いHiRDBシステムをスムーズに実現するための技術支援サービスであり、次のメニューを用意している。

■ DB設計支援サービス

データベースの論理設計情報（ER図）を元に、HiRDB表やインデックスの物理設計を行ったり、レビューに参加して改善策を提案する。

■ DBシステム設計支援サービス

各種容量見積りやRDエリア設計、システム定義パラメタ作成、ファイルシステム設計を行ったり、レビューに参加して改善策を提案する。

■ DB運用設計支援サービス

データベース業務要件書やシステム構成設計書、DB設計書などの情報を元に、日々のシステム監視方法やバックアップ方式、障害時の対応方法など最適な運用方法を提案したり、運用手順書を作成する。

■ データ連携設計支援サービス

HiRDB DatareplicatorやHiRDB Dataextractorを使ったデータベース間のデータ連携について、方式設計やシステム定義パラメタ作成を行ったり、レビューに参加して改善策を提案する。また、オフサイトバックアップ（広域災害対策）に向けたデータ連携方式の提案もする。

■ DB構築支援サービス

HiRDBのインストールをしてからサンプル表をアクセスできるようになるところまでの環境構築を行う。また、糸切り替え環境構築、バージョンアップ、システム構成変更も行う。本サービスでは、HiRDBの導入、バージョ

ンアップ、システム構成変更などの各手順書の作成に必要な情報を提供する。

■ DB移行支援サービス

他DBMSからHiRDBへの移行やサーバ機入れ替え時の移行をスムーズに行うため、アプリケーションプログラム、データベース、定義（ディクショナリ）の移行方法についてコンサルテーションをする。また、データベースや定義の移行作業を行う。

■ 性能チューニング支援サービス

HiRDBシステムのチューニングアップを支援する。SQLやバッファ制御などについて、性能分析資料の取得・分析方法から改善策を提案する。また、取得・分析作業の代行や分析結果のレビューも実施する。

■ 教育サービス

HiRDBの概要からシステム設計、運用、アプリケーションプログラム開発、チューニングなどの講義と演習を通じて、HiRDBに関する知識を身に付けるための教育で、オーダーメイドが可能。

5.2 | リーズナブル&スケーラブルな価格

HiRDB Version 6では、製品そのものの低価格化を行うと同時に、シンプルかつネットビジネスに最適なライセンス体系に刷新されている。その具体的な刷新内容は、以下の通りである。

- ・従来からの同時接続数ライセンスに加えてCPU数ライセンスを導入
- ・プラットフォームフリーでの価格（UNIX/Windows®/Linuxで同一価格）
- ・Single ServerとParallel Serverの価格差解消
- ・ライセンス単価の設定による不要なライセンス数の解消

HiRDB Version 6でのサーバ製品標準価格は、以下の通りである。

- ・Single Server及びParallel Server
 - －12万円／1同時接続¹⁾
 - －180万円／1CPU

・ Workgroup Server

—2万6千円／1同時接続²

—95万円／1CPU³

*1：最小同時接続数は、Windows[®]版とLinux版では2、UNIX版では4

*2：最小同時接続数は2

*3：CPU数の上限は4

これにより、システム導入コストを最小に抑えることができることに加え、従来からの上位互換性と長期サポートは健在なので、システム維持コストも極小化できる。

HiRDB Version 6は、製品アーキテクチャのスケールビリティを活かす価格とライセンス体系になっており、「小さく始めて大きく育てる」ネットビジネスにベストマッチと言える。

5.3 | 基本仕様

■ 基本仕様

サポートSQL レベル	SQL92 エントリレベル準拠、SQL99 ベース (オブジェクト指向拡張)
API	SQL、SQL/CLI、ODBC 2.0、JDBC 2.0、OLE DB 1.1
最大列数／表	30,000
最大行数／表	無制限 (システムリソースに依存)
最大行長	無制限 (システムリソースに依存)
最大クライアント数	無制限 (システムリソースおよびユーザライセンス数に依存)
最大DB サイズ	1,024PB
排他制御単位	表、ページ、行
ストアプロシージャ	ISO国際標準準拠
並列ユティリティ	データベース作成、バックアップ、リカバリ、データベース再編成
DB分割方式	キーレンジ分割、ハッシュ分割、フレキシブル・ハッシュ分割
並列アーキテクチャ ⁴	シェアドナッシング型
並列SQL ⁴	SELECT、UPDATE、INSERT、DELETE

*4：HiRDB Parallel Serverの仕様

■ データ型

整数	SMALLINT、INTEGER
固定小数点数	DECIMAL (29桁)
浮動小数点数	SMALLFLT、FLOAT
固定長文字列	CHAR、NCHAR、MCHAR
可変長文字列	VARCHAR、NVARCHAR、MVARCHAR

バイナリ	BLOB (ISO国際標準準拠)
日付、時刻	DATE、TIME
日間隔	INTERVAL YEAR TO DAY
時間隔	INTERVAL HOUR TO SECOND
論理値	BOOLEAN
ユーザ定義	抽象データ型

■ 対応プラットフォーム (サーバ製品)

- ・ HP-UX
- ・ Solaris
- ・ AIX
- ・ HI-UX/WE2
- ・ Linux
- ・ Windows NT[®]
- ・ Windows[®] 2000

■ 対応プラットフォーム (クライアント製品)

- ・ HP-UX
- ・ Solaris
- ・ AIX
- ・ HI-UX/WE2
- ・ Linux
- ・ Windows[®] 95
- ・ Windows[®] 98
- ・ Windows NT[®]
- ・ Windows[®] 2000

商標表記

- ・ AIXは、米国における米国International Business Machines Corp.の商品名称です。
- ・ Java[™]およびすべてのJava関連の商標およびロゴは、米国及びその他の国における米国Sun Microsystems, Inc.の商標または登録商標です。
- ・ ODBCは、米国Microsoft Corp.が提唱するデータベースアクセス機構です。
- ・ OLEは、米国Microsoft Corp.が開発したソフトウェア名称です。
- ・ JBuilderは、米国および、その他の国における米国Borland Software Corporationの商標です。
- ・ ORACLEは、ORACLE Corporationの登録商標です。
- ・ PC/ATは、米国International Business Machines Corp.の商品名称です。
- ・ PowerBuilderは、米国法人Sybase, Inc.の登録商標です。
- ・ Red Hatは、米国Red Hat software, Inc.の登録商標です。
- ・ Solarisは、米国Sun Microsystems, Inc.の登録商標です。
- ・ SPARCstationは、米国Sun Microsystems, Inc.の登録商標です。
- ・ TurboLinuxの名称及びロゴは、TurboLinux, Inc.の商標です。
- ・ UNIXは、X/Open Company Limitedが独占的にライセンスしている米国ならびに他の国における登録商標です。
- ・ VERITASおよびVxFSは、米国VERITAS Software Corporationの登録商標です。
- ・ Visual C++は、米国およびその他の国における米国Microsoft Corp.の登録商標です。
- ・ Windows[®]は、米国およびその他の国における米国Microsoft Corp.の登録商標です。
- ・ Windows NT[®]は、米国およびその他の国における米国Microsoft Corp.の登録商標です。
- ・ X/Openは、X/Open Company Limitedの英国ならびに他の国における登録商標です。

その他記載の会社名、製品名は、それぞれの会社の商標もしくは登録商標です。