

## 知的分散データベースにおけるデータ管理方式

永瀬恵子 Andrew Howells 原嶋秀次

(株) 東芝 研究開発センター  
システム・ソフトウェア生産技術研究所

知的分散データベース (IDPS-DB) は近年のシステムに対する柔軟性、信頼性、拡張性の要求を、基本システムの1つであるデータベースにおいても実現するために研究・開発中の分散データベースである。本データベースは現在、データとしてリレーションを扱っていて、その管理の方式として2つの方式を持っている。1つはリレーションの高度なデータ独立性を実現する方式、もう1つは、リレーションの集合をグループ化して管理する方式である。

本稿では、この2つのデータ管理方式について説明し、さらに、リレーションをグループ化して管理する方式を用いて、意味に基づいてグループ化を行いデータベースを構築することによりデータの管理やアプリケーションの処理が容易となることを示す。

### Data Management Mechanisms of IDPS-DB

Keiko Nagase, Andrew Howells and Shuuji Harashima

TOSHIBA Corporation Research & Development Center  
Systems & Software Engineering Laboratory

70, Yanagi-cho, Saiwai-ku, Kawasaki, Kanagawa, 210 JAPAN

IDPS-DB is a distributed database system under development in order to realize an adaptable, reliable and extensible system. This paper describes the data(relation) management mechanisms of IDPS-DB. IDPS-DB has two such mechanisms - one is for realizing high data independence, and the other allows grouping of relations. This paper also describes an example of database construction based on the second mechanism, and shows the merits of IDPS-DB for use in data management and application processing.

## 1. はじめに

計算機を用いたシステムは、応用分野の広がりと共に機能の高度化が進み、大規模かつ複雑になってきている。同時に、その役割はますます重要になってきていて、システムの停止などによる社会への影響は重大である。

このような状況において有効なシステム構成法の研究が各所で行われている。独立した複数の計算機が通信回線で接続され時々変化する状況において、各計算機が協調してそれぞれのもつ機能を提供しながら求められる処理を遂行する協調分散システムは、柔軟性、信頼性、拡張性にすぐれ、将来のシステム構成法として注目されている<sup>[1]-[3]</sup>。

東芝では、この考え方をソフトウェアの構成に適用した知的分散システムを提案している<sup>[4][5]</sup>。知的分散システムでは、システムを構成するソフトウェア要素（これをオブジェクトと呼ぶ）がそれぞれのもつ機能を提供しながら必要な処理を遂行する。オブジェクトは通信回線で接続された複数の計算機上に実装され、必要に応じて移動したりコピーを他の計算機に持つことが可能であり、より柔軟で信頼性があり、拡張性に優れたシステムを構成することが可能である。

我々は現在、知的分散システム上でデータベースの研究・開発を行っており、これまでに、リレーションを自身のデータとして保持するオブジェクトで構成するデータベース（知的分散データベース）を開発している<sup>[8]-[10]</sup>。このデータベースでは、各オブジェクトは複数のリレーションを管理し、それらのリレーションに対する処理を実行する。また、結合のように複数リレーションに関係する処理では必要に応じて他のオブジェクトと協力しながら処理を実行する。リレーションを管理するオブジェクトは複数の計算機にコピーを置くことが可能であり、その際にはコピーを用いた並列検索などを自律的に実施する。また、各オブジェクトの動作は計算機の構成の変化に影響を受けないので、新たな計算機の追加や削除などを

自由に行うことが可能である。

本稿では、この知的分散データベースについてその構成と機能を紹介する。

第2章では、知的分散データベースのプラットフォームとなる知的分散システムについて述べる。

第3章では、本データベースを開発した背景について述べる。

第4章では、本データベースの知的分散システム上での構成、特にデータ（リレーション）の管理方式について述べる。

第5章では、本データベースの機能と第4章で述べたリレーションの管理方式を用いたデータベースの構築例について述べる。

## 2. 知的分散システムの概要

知的分散システムは、集中管理機構を排した分散システムのアーキテクチャである<sup>[4][5]</sup>。自律的で互いに協力・協調するシステム要素群によって構成され、柔軟性、拡張性、信頼性の高い分散システムの実現を可能とする。

知的分散システムは現在、LANによって接続された計算機（サイト）上に構築されており、システム要素はデータとそれに対する操作の集合であるメソッド（手続き）群から成るオブジェクトとして実現する。オブジェクト群を管理する集中機構を持たないので、システム全体の状況の保持、管理を行なうようなサイトやシステム要素は存在しない。オブジェクト間の同期や排他制御、負荷分散や機能配分などシステム全体に渡る管理はオブジェクト間の協力・協調によって行なわれる。

このようにオブジェクトが自身の機能の実現に必要な機構だけでなく、他オブジェクト群と協力・協調する機構を持つので、オブジェクトの追加や変更、削除、サイト間移動などが局所的な操作で実行でき、柔軟性、拡張性、信頼性の高いシステムの実現が可能となる。

ここで、協力とは与えられたジョブの処理をオブジェクト群が互いに処理結果をやりとりすることにより遂行することであり、協調とはオブジェ

クト間で処理の競合を調整し、システム資源の有効かつ正しい利用（排他管理、負荷及び、機能の配分管理）をはかることを意味する。

図1に知的分散システムにおけるオブジェクトの構成を示す。これらのオブジェクトは自律的かつデータ独立である。オブジェクトが自律的であるとは、自身の行動規範を自身に有し、かつその行動方法（すなわち手続き）を自身に有していることを、またデータ独立であるとは、その行動規範が外部から隠ぺいされており、外部からは決められたインタフェースを通してしかアクセスできないことを意味する。これらに必要な知識は、共通知識と固有知識とに分けることができる。固有知識はオブジェクトに固有な性質を表わすデータと手続き群から、共通知識は排他制御、故障管理、負荷分散などのシステム全体の管理に必要なデータと手続き群から成る。前述したオブジェクト間の協力・協調は、各オブジェクトの持つ共通知識の利用によってなされる。

システムに与えられたジョブは、前述したようにオブジェクト間の協力・協調によって処理される。この間の協力・協調に際しては、「どこに、どんなオブジェクトがどれだけあるか」といった情報が必要となるが、知的分散システムでは必要になった時点で個々のオブジェクトが放送通信を用いて自律的に他のオブジェクトからこれらの情報を得るので、オブジェクトの位置や多重度を意識する必要がない。

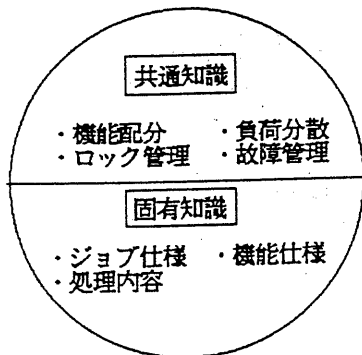


図1. 知的分散におけるオブジェクトの構成

### 3. 知的分散データベース開発の背景

前章で述べたように、知的分散システムは柔軟性、拡張性、信頼性などの向上を目的として提案した分散システムのアーキテクチャである。このようなシステムは各所で必要とされるものである。

一方、システムが複雑になり、対象とするデータが増えれば増えるほどシステムの基本ツールの一つであるデータベースはその重要性を増し、柔軟性や拡張性、信頼性といった一般システムと同様の特徴が求められている<sup>[6][7]</sup>。我々はこの様な要求に答えるために知的分散データベースの開発を進めてきた<sup>[8]-[10]</sup>。

本データベースは、リレーショナルデータを管理する自律的なデータオブジェクト群とその他のシステム管理オブジェクトによって構成される。データオブジェクトは、自身の管理するリレーションに対する処理要求に対してなすべき処理を判断し必要な処理を実行する。したがって、本データベースシステムには従来のような集中的なデータベース管理システム(DBMS)は存在せず、その機能は自律的なオブジェクト間の協調動作によって実現される。これにより、

1. サイト構成が変化しやすい
2. 利用可能なデータが動的に変化する

など、アプリケーションが現時点でどのデータ（リレーション）にアクセスすれば必要な処理が実行されるのか判断できないようなシステムにおけるデータベースとして有効である。すなわち、アプリケーションが実行したい処理の内容だけを伝えれば、本データベースを構成するオブジェクト群の自律性により必要な処理を行うことが可能である。

### 4. 知的分散データベースのシステム構成

ここでは、本データベースの知的分散システム上での構成、特にリレーションの管理方式について

て述べる。

図2に本データベースの構成を示す。

リレーションを管理する「データオブジェクト」、アプリケーションオブジェクトから見た本データベースへのアクセス窓口となる「質問処理オブジェクト」、各サイトの状況を把握する「サイトマネージャオブジェクト」、実際に本データベースを利用する「アプリケーションオブジェクト」から構成される。

データオブジェクトは、リレーションの管理の仕方によって2種類存在する。図3にそれぞれのデータオブジェクトを示す。

(a)の構成の(論理的な)データオブジェクトでは1つのデータオブジェクトは1つのリレーションを管理する。このデータオブジェクトは、実際にはファイル処理オブジェクトとリレーションオブジェクトと呼ばれる3層スキーマの概念の内部スキーマおよび概念スキーマに対応する2つのオブジェクトで実現されており、高度のデータ独立性を達成している。すなわち、ファイル処理オブジェクトはデータファイルを管理し、リレーションオブジェクトにファイルとしてのスキーマを提供する。リレーションオブジェクトは、ファイル処理オブジェクトが提供するファイルとして

のインタフェースを利用して質問処理オブジェクトにリレーションとしてのスキーマを提供する。このような構成をとることにより、様々なファイルシステムや既存のリレーショナルデータベースを、それらを扱うファイル処理オブジェクトまたはリレーションオブジェクトを作成することで統合的に利用することが可能となる。この方式は知的分散データベース開発当初より採用してきたものである[8]-[10]。

一方、その後、(b)の構成のデータオブジェクトのように1つのデータオブジェクトが複数のリレーションを管理する方式の開発を行った。この構成では、1つのデータオブジェクトが複数のリレーションの内部スキーマを直接ファイルのアクセスを行って管理するとともに、それらのリレーションの概念スキーマをも管理する。したがって、(a)の構成に比べてデータ独立性の点で劣っているものの、1つのオブジェクトで管理されるリレーション集合の単位でのデータ独立性は保たれていて、これは多くの状況において十分なデータ独立性であると考えている。

このようなリレーションの管理方式を採用したのは、以下のような理由による。

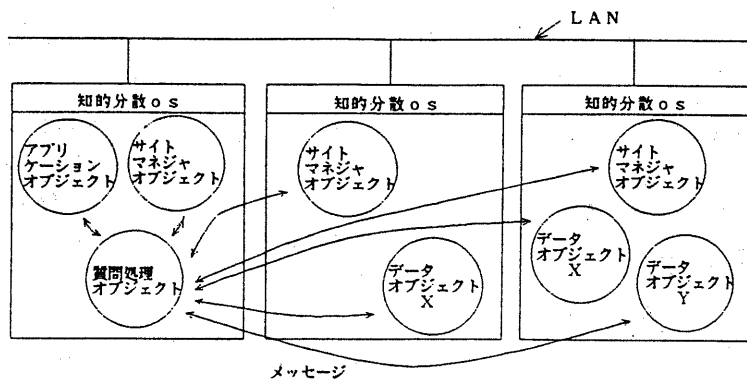
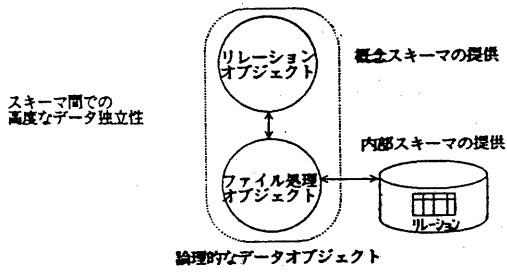


図2. 知的分散データベースの構成



(a) 2オブジェクトによる構成



(b) 1オブジェクトによる構成

図3. データオブジェクトの構成

- (1) 意味的にまとまりのあるリレーションの集合を効率よく扱いたい。
- (2) 意味的にまとまりのあるリレーション間で、操作の制約などアプリケーションに応じた処理を簡単に実現したい。
- (3) システムを効率よく動作させたい。

(1) はデータベースに存在するリレーションの中で、ある意味的なまとまりを持つリレーションの集合をまとめて、それぞれ別のデータオブジェクトで管理することで実現する。これにより、従来のデータベースにおいて、横並びの関係にあったリレーション群に一種のビューの機能を取り入れることが可能となり、この集合の定義の仕方によって、5章で後述するような効果的なデータベースの構築が可能となる。

(2) は意味的なまとまりのあるリレーションの集合を管理するデータオブジェクトに各リレーションのスキーマを提供する機能だけでなく、そのリレーションの集合をアクセスするアプリケーションに応じた処理を持たせることによって、ア

プリケーションの処理を容易にする。

(3) はメッセージ交換に関するオーバーヘッドを削減したいという要望による。すなわち、(a) の論理的なデータオブジェクトは、実際は2つのオブジェクトで構成され、それらの情報のやり取りは全てメッセージ形式で行われる。したがって、これらのオブジェクトを1つにまとめることにより、メッセージ交換によるオーバーヘッドを除くことができ、システムの効率が向上する。また、1つのデータオブジェクトが複数のリレーションを管理するので、結合演算の場合に対象となるリレーションが同じデータオブジェクトで管理されていると結合演算がより高速に処理可能となる。

さらに、他のオブジェクトについても簡単に説明する。

質問処理オブジェクトは、アプリケーションからの質問を受け付け、その時点でのシステムの状態を確認して処理戦略を立て、本データベースを構成する各オブジェクトに必要な処理を依頼するオブジェクトであり、アプリケーションと本データベースのインタフェースとなる。したがって、質問処理オブジェクトが質問処理言語を提供する。

処理戦略については、質問を受け付けた時点でのサイト数、それぞれのサイトの負荷、それぞれのサイトに存在するオブジェクト（すなわちリレーション）などを後述するサイトマネージャオブジェクトに問い合わせて調べ、受け付けた質問を実行するために必要な処理を各データオブジェクトに依頼する。その際、コピーの存在する（多重化された）データオブジェクトがあれば、検索範囲を分割し、複数のコピーに並列検索を依頼して処理の高速化を行うことも可能である。

サイトマネージャオブジェクトは、各サイトに1つ存在し、自身が存在するサイト（自サイト）の状況（サイトの負荷、存在するデータオブジェクト）を把握している。そして、上述の質問処理オブジェクトからの要求に応じて自サイトの状況を返答する。

	COUNTRY	MAKER	NAME	CLASS
11	JAPAN	ISUZU	GEMINI ZZ-R	1800
12	U S A	CORVETT	ZR-1	0
13	G B	LOTUS	SUPER 7	0
14	JAPAN	enfini	RX-7 Type R	2000
15	GERMANY	B M W	2002	2000
16	ITALY	ALFA ROMEO	1750 GT	1750
17	JAPAN	NISSAN	SKYLIFE 2000 GTR	2000
18				

図4. リレーション操作画面

アプリケーションオブジェクトは、それぞれのアプリケーションごとに作成され、上述した3種類のオブジェクトによって構成される本データベースを利用してユーザの必要な処理を実施するオブジェクトである。本データベース用の汎用アプリケーションとしてリレーション操作のMMIを開発済みである。図4に代表的な操作画面の例を示す。

## 5. 知的分散データベースの特徴

本章では、本データベースの機能として、関係演算の処理方法を示し、さらに本データベースの特徴を生かしたデータベースの構築例を示す。

### 5.1 関係演算

本データベースでは、上述したように信頼性の向上などを目的としてデータオブジェクトのコピーを複数サイトに配置する多重化が可能である。このようにデータオブジェクトが多重化されている場合、本データベースでは多重化されたデータオブジェクトが管理するリレーションの検索範囲を分担して並列検索を行い、信頼性の向上だけでなく処理効率の向上を実現している。

以下、リレーションの多重化を利用した並列検索について説明する。

図5に多重化されたリレーションRの並列検索の概要を示す。図5では簡単のためリレーションRを管理するデータオブジェクトAは図3の(b)の構成とし、さらに各データオブジェクトAはリレーションRのみを管理するものとする。まず、検索要求がアプリケーションから質問処理オブジェクトに与えられる。質問処理オブジェクトはシステム中のデータオブジェクトAへリレーションRの多重度を放送通信を用いて問い合わせ、得られた答からリレーションRの多重度を判断する。そして、インデックスが利用できずリレーションのタプルを全て調べる必要がある場合には(インデックスが利用できる場合には多重化を利用しない方法を採用している)、リレーションRの検索範囲を $n$ 等分して(各サイトの負荷は等しいとする)、多重化された各データオブジェクトAにそれぞれ検索範囲を通知して、リレーションRの検索を依頼する(図5(a))。各データオブジェクトAはそれぞれリレーションRの検索を並列に行い(図5(b))、その結果を質問処理オブジェクトへ通知する。質問処理オブジェクトは通知された結果を集計して最終的な結果を求め、その結果をアプリケーションへ通知する(図5(c))。

表1に図3の(a)の構成で、データオブジェクトを1サイトに配置した場合の検索演算の性能を示す。この結果から、本システムの性能面での実用性は高いと考えている。なお、測定では、各

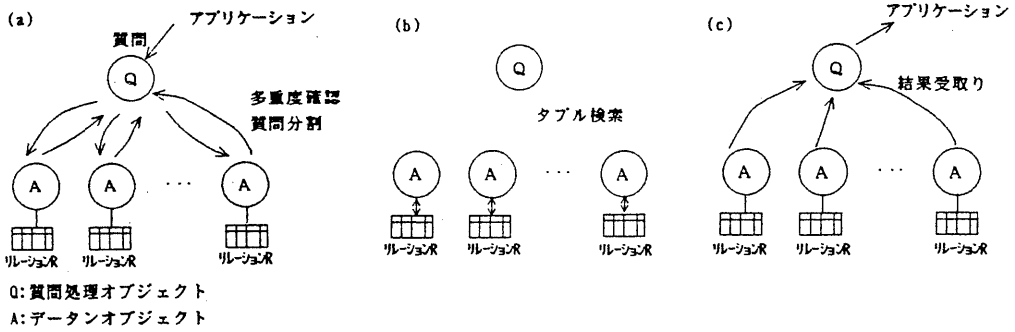


図5. リレーションの並列検索

サイトに小型UNIXワークステーションを用い、通信回線としてイーサネットを用いている。現在のところ、多重化による効果は評価していないが、別のマシンを用いた論理的なデータオブジェクトを多重化した場合の性能向上についてはすでに報告済みであり<sup>[9]</sup>、ここでも同じ効果が期待できると予想される。この他、本データベースでは結合、射影などの関係演算についてもデータオブジェクトの多重度を利用して並列演算を実施する。

また、図3の(b)の構成のデータオブジェクトを利用した場合には、4章で述べた理由から、より高速な処理が可能となる。これらの詳しい性能測定に関しては今後実施していく予定である。

## 5.2 知的分散データベースを利用したデータベース構築例

ここでは、ある意味付けによりリレーションの集合をグループ化して1つのデータオブジェクトで管理してデータベースを構築することにより、アプリケーションの処理が容易になることを示す。

図6はリレーションをグループ化してデータベースを構築した例である。データオブジェクトとして、データオブジェクトA、B、Cがある。データオブジェクトAは自身の内部でa、b、cと名付けられたリレーションを管理し、データオブジェクトBは同様に自身の内部でa、b、cと名付けられたリレーションを管理しているとする。ここで、データオブジェクトAとデータオブジェクトBが管理するリレーションa、b、cはそれぞれ実体は別のものであるが、個々のデータオブジェクトの中では同じ意味を表すものとして同じ名前で管理することが可能である。

このように同じ意味を持つリレーションをグループ化することで、グループ内のリレーションは他から独立となり、その管理が容易になる。また、データオブジェクトごとにそのグループ内のリレーションの集合に特有の処理を定義することも可能となる。

さらに、データオブジェクトCは様々なアプリ

マシン	J3100SGT	AS4630
0%	0.3秒 25.3秒	0.01秒 0.4秒
1%	3.7秒 28.5秒	0.8秒 1.3秒
10%	25.3秒 48.3秒	10.0秒 12.1秒
20%	48.9秒 72.3秒	19.3秒 19.6秒

注1) データ件数 = 10,000  
注2) 上段: インデックスあり  
下段: インデックスなし

表1. 単一マシンでの検索演算の性能

ケーションが共通に用いるリレーションx, yを管理しているとし、アプリケーション1はデータオブジェクトAとC、アプリケーション2はデータオブジェクトBとCを利用するアプリケーションとする。ここで、リレーションx, yは複数のアプリケーションに共通に用いられるデータなので、1つのデータオブジェクトCで管理するようにする。

一般に、データベースには様々なユーザ（アプリケーション）によって共通に利用されるリレーションが管理されているが、このように利用の共通性によってグループ化を行うと、実際のデータの変更がおこった場合などにどのリレーションのデータを変更すれば良いのか、またその影響はどのアプリケーションに及ぶのか、などを簡単に知ることができ、データの管理が容易になる。

以上のように、ある意味をもった複数のリレーションの集合ごとに1つのデータオブジェクトで管理してデータベースを構築すると、従来のリレーショナル・データベースよりも、データ管理やアプリケーションの作成を容易にすることが可能である。そして、このデータベースは様々な応用分野に適用可能と考えている。

## 6. おわりに

本稿では、知的分散データベースとそのリレーションの管理方式について、その概要を紹介した。

リレーションを管理する自律的なデータオブジェクトの集合でシステムを構成することにより、信頼性、柔軟性、拡張性に優れたデータベースを実現することができ、また意味のあるリレーションの集合を1つのグループとして管理することで、アプリケーションの作成が容易になるようなデータベースの構築が可能なることを示した。

今後、本システムにおいて

- ・ 質問処理言語
- ・ 障害時処理
- ・ 処理効率の向上
- ・ トリガなどアクティブ機能の導入

などについての研究・開発をおこなう予定である。

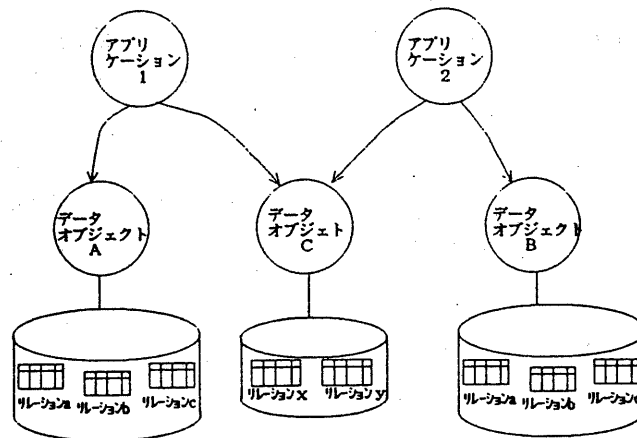


図6. データベース構築例



## 参考文献

- [1] 長田 : "自律分散システムと情報処理,"  
計測と制御誌 Vol.29 No.10 pp.935-938  
(1990-10).
- [2] 森 : "自律分散システムと制御分野での実用例,"計測と制御誌 Vol.29 No.10  
pp.923-928 (1990-10).
- [3] 中島 他:"自律分散システムに関する一考察  
—大規模ネットワークにおけるネットワークサービス提供のための基盤モデル—,"  
情処学ソフトウェア基礎論研資27-4  
pp.33-40 (1988.12).
- [4] 関 他:"知的分散OS,"  
情処学マルチメディア研資 (1988-9).
- [5] 田村 他:"知的分散システムのアーキテクチャ,"電学誌 108-C[6](1988-6).
- [6] 宇田川:"高水準データモデル,"  
情処学誌 Vol.32 No.9 pp.1008-1013  
(1991-9).
- [7] 増永:"次世代データベースシステムとしてのオブジェクト指向データベースシステム,"  
情処学誌 Vol.32 No.5 pp.490-499  
(1991-5).
- [8] 原嶋 他:  
"知的分散データベースシステム,"  
情処学情報学基礎研資 (1988-11).
- [9] K.Nagase et. al. :  
"IDPS Database System,"  
Proc. of JWCC-6 pp.99-106 (1991-7).
- [10] Andrew Howells et. al. :  
"Partial Queries in a Dcentralised  
Distributed Relational Database,"  
Proc. of Future Database '92 pp.97-105  
(1992-4).