

解 説

データ・ディクショナリ／ディレクトリの動向*

酒 井 博 敬**

1. はじめに

データ・ディクショナリ／ディレクトリ (DD/D=`data dictionary/directory`) は、メタ・データの集中貯蔵所である。メタ・データとは、データの属性に関するデータの意味で、データの名前、表現、構造、所在、使い方、さらに、データの意味、発生源、収集責任者、使用権、有効期限などを指す。そして、メタ・データを管理するシステムが DD/D システムである。

DD/D システムは、情報システム、とくにデータベース・システムにおける運用管理手段として要請され、発展してきた。現在ではその体系化が進み、DD/D はデータベース管理システム (DBMS=`data base management system`) に統合される傾向にある。

2. データベース・システムと DD/D

メタ・データはデータベース固有のものではない。また計算機に記憶されたデータに限られたものでもない。それは、通常非データベース・ファイル、また帳票や資料を利用する上で必要とされてきたものである。そして、ドキュメント、制度、人の記憶など、さまざまな形で管理されてきたものである。データベースが導入され、実用に供されるようになって、メタ・データ管理の組織化が促されることになったといえよう。その必然性は、データベース・システムのもつ次の特性から伺われる。

- (1) データベースの設計は、個々のデータ要素、それを利用するユーザのクラス、およびシステムが提供するサービスについて、それぞれの属性と相互関係を調査分析して決定される。
- (2) データベースの編成とアプリケーション・プログラムの仕様は、個々のユーザ要求単位ではなく、データ要素に対して施されるべき共通機能の集

約、および性能、完全性 (integrity)、安全性 (security) などの維持を配慮して設計される。だから、必要なデータが、どこに、どんな形で蓄積され、どのプログラムが要求を実行するのか、また、どのユーザは、どのプログラムとどのデータをアクセスすることが許されるのかを、ユーザに対して明確にしなければならない。

(3) データベースという容れものが用意されても、そこに蓄積されるべき正しいデータが収集されねばならない。そのためには、データの発生源、収集責任者、データの正当性を確認する基準などを定め、管理しなければならない。

(4) データの属性やアプリケーション・プログラムの仕様を変更すれば、それは他のデータやプログラムにさまざまな影響を及ぼす、システム要素の属性変更は、その波及効果の完全な評価を必要とする。

以上にあげたものはすべて、メタ・データを対象とした分析、操作および維持の作業を必要とするものであり、データベース管理者が果たすべき機能とされているものである。しかし、プログラム数が 100 本あればメタ・データは数万になるといわれる。計算機におけるデータ対象の処理と同期をとって、対応するメタ・データを人間が維持すること自体、大きな困難をとまなうわけである。メタ・データの管理がデータベース管理者の能力限界を越えたとき、DD/D システムはシステム管理の基本手段として自然に要求されるようになったものである。

ところで、DD/D という用語に文字どおり解釈を与えると、次のようになる。

まず、データ・ディクショナリは、データのソース定義ライブラリである。つまり、データの論理および物理構造、入力データの正当性仕様、出力編集仕様、符号化、非符号化の仕様、さらにユーザの理解を助けるためのテキスト記述などを蓄積したものである。

データ・ディレクトリは、データのオブジェクト定義ライブラリである。これは、データ・ディクショナ

* The Trend of Data Dictionary/Directory by Hiroataka SAKAI (Software Works, Hitachi, Ltd.)

** (株)日立製作所ソフトウェア工場

表1 DD/D システム・パッケージ

パッケージ名	ベンダー
DATA BASE DIRECTORY	Eastern Air Lines, Inc.
DATA CATALOG	Synergetics Corporation
DATA DICTIONARY	British Rail
DATAMANAGER	Management Systems and Programming, Ltd.
IMS DICTIONARY SYSTEM	IBM Corporation
LEXION	Arthur Andersen & Co
PRIDE LOGIK	M. Bryce & Association, Inc.
'THREE D' SYSTEM	Anderson Clayton Foods
UCC TEN (DATA DICTIONARY/MANAGER)	University Computing Company

(アルファベット順)

り内のソース定義の中で計算機に必要な部分を機械コードに変換し、データの物理的な所在場所やアクセス制御情報を付加したものである。

これから分かるように、ディクショナリはユーザによる使用を、またディレクトリは計算機による使用を目標としている。両者は、本来同一の実体の属性を、目的別に異なる形に表現したものである。

現在、表-1 にあるような DD/D システム・パッケージが提供され、主としてデータベース・ユーザで使用されている。いずれもデータ・ディクショナリ機能だけを扱い、データ・ディレクトリの維持は DBMS あるいはオペレーティング・システムにゆだねている。

これら DD/D システムに見られる目標をあげるならば、次の3点に要約できる。

- (1) データの収集、蓄積、処理、および供給において、不用意に発生する冗長性と矛盾性を防ぐ。
- (2) アプリケーション・システムの開発と変更にかかる時間と費用を節減する。
- (3) データの使用と責任に関する標準の設定と実施を容易にする。

これらは、本来 DBMS の目標とされているものであるが、DD/D システムは DBMS との併用によって、この目標をより広い範囲にまで拡大しようとするものである。

3. DD/D の内容と構成

GUIDE はユーザ・ニーズの詳細な調査にもとづいて、1975年、DD/D への要望¹⁾をまとめた。この中に、データベース・システムにおいて必要とされるメタ・データが分類されている。有用な情報と思われるので、以下にそのリストをあげる。

(1) 識別属性

識別番号、データ名、キーワード表示子(キーワー

ドのリストからなる成句)、テキスト記述(意味、用途、共通名など)、型(名前、コード、カウント、量、日付、テキストなど)、機能(アプリケーションにおいてデータが果たす役割、連結、識別、集計、制御、インデックス、順序など)、版、状態(提案、同意、承認、有効、廃止など)、有効期限、世代、分類(エレメント、グループ、レコード、ファイル、データベース、プロセス、発生源、出力、別名、非データベース単位など)

(2) 表現属性

長さまたはサイズ、キャラクタの型(ビット・ストリング、キャラクタ・ストリング、数値データなど)、精度、記録方式(固定長、可変長の別)、けた位置(左づめ、右づめなど)、コード構造(BCD、EBCDIC、USASCII など)、ピクチャ、編集規則(定数、領域、編集マスク、テーブルなど)、スケール、デフォルト値、空値表示、単位、誘導アルゴリズム(データ単位を他のデータ単位から導くためのアルゴリズム)、フォーム・エクステント(同じ蓄積レコード記述をもつレコードが蓄積されるデータ・セットの区域)

(3) ロケーション属性

装置の型、ボリューム識別、編成(順編成、乱編成、リスト編成、転置編成など)、アクセス法(順アクセス、直接アクセス、索引順アクセスなど)、番地づけアルゴリズム(ランダムマイジング・アルゴリズム、あふれに対する解決アルゴリズムなど)、物理番地、ブロック・サイズ、物理配列順、ディレクトリ別名(インデックス、ボリューム・テーブルなどに蓄積されるデータに対して使われる名前)

(4) 関係属性

別名または同意語、論理配列順、従属関係(データ単位がより上位の論理単位に属していることの記述)、相互作用(関係する他のデータ単位の定義)、関係の型(データ単位間で維持される関係の型・配列順、ファイルに対するキー、親子関係、包含関係など)

(5) 編成属性

実現値(occurrence)の期待値および最大値、成長率(実現値数の予想増加率)、アクセスおよび更新頻度、あふれ(データ単位が実現値数の最大値を越えたときにとられるべき処置)、優先度(データベース内の資源あるいはサービスに対する競合を解決するための、データ単位の相対ランクづけ)、統計記録(種類、時期、方法など)

(6) 安全性属性

パスワード、安全性のクラス（データ単位の安全性レベルの分類。ユーザのアクセス権は安全性のクラスに対応して設けられる。）、アクセスの型（ある安全性レベルをもつユーザに対して許されるアクセスの種類。検索のみ、更新など）、データ発生源、更新責任、定義責任、アクセス認可権、アクセス権別ユーザ・リスト、特殊扱い（符号化、暗号化など）、パスワード変更アルゴリズム、パスワードおよび安全性キーの変更頻度、試行回数制限（データ単位への不当アクセスが試みられたとき、防御措置がとられる前に許容される試行回数の制限）

(7) 完全性属性

正当性（データ単位の値を受け入れる前にとられるべき確認事項、キャラクタの型、けたチェック公式など）、無矛盾性（データ単位またはその集合の値を受け入れる前になされるべき無矛盾性チェックの指定）、合理性（データ単位の値を受け入れる前になされるべき合理性チェック。たとえば値域チェック）、波及集合（データの無矛盾性維持のために、同時に更新、変更、あるいはロックされるべきデータ単位の集合）、完備性（上位のデータ単位を受け入れるために存在しなければならない、関連するデータ単位集合の指定。たとえばレコード内の必要データ項目）、信頼性（データ単位をどの程度信頼できるかの指定。見積り値、期限遅れ、正当性チェック不適合など）、回復（データ単位の更新誤り、消失などの際にとられるべき回復措置）、保存期間

さて、メタ・データ管理の完全性を期するためには、DD/D は情報システムのすべての要素に関する広範な定義情報を含んでいることがのぞましい。つまり、データだけでなく、プロセス、トランザクション、レポート、発生源、ドキュメント、ユーザなどの実体を対象として、これら実体が本来もっている属性、およびその環境要素との関係において定まる属性のすべてが、DD/D に蓄積されなければならない。ここでは、簡単のために次のシステム要素をとりあげて、DD/D の構成を考えてみる。

- エレメント (EL): プロセスによって参照されるデータの最小単位。
- グループ (GR): 論理的に関連するエレメントの集合。データに対するユーザの見方に応じて異なるグループが存在する。
- 物理セグメント (PS): 外部ストレージに蓄積

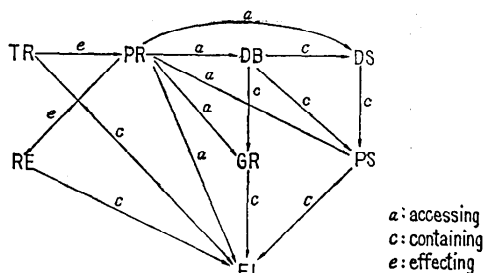


図-1 メタ・データ・グラフ G0

されるデータの最小アクセス単位。

- データ・セット (DS): 外部ストレージ上の PS の集合。
- データベース (DB): 関連するデータ・セットの集合。
- プロセス (PR): あるデータ処理タスクを遂行する手続き。
- トランザクション (TR): あるプロセスを実行するトリガーとなる入力データの集合。
- レポート (RE): システムからユーザへ出力される情報。

これら実体に関するメタ・データは、図-1 のようなグラフ G0 に表現される。グラフの節は実体自身に関するメタ・データ、枝は実体間関係に関するメタ・データを表す。枝の方向とラベルは、成分関係 (c=containing)、使用関係 (a=accessing)、因果関係 (e=effecting) などを示す。

グラフ G0 はメタ・データの一般的な構造を表すものであるが、個々の実体の部分集合に関するメタ・データを、G0 の下位のグラフとして表現することができる。次はその例である。

(例1) 「物理セグメント PS1 はエレメント EL2 と EL3 からなる。エレメント EL1 と EL2 からなるトランザクション TR1 は、プロセス PR1 を起動する。PR1 は PS1 を更新モードでアクセスし、EL2 を更新する。」この記述は図-2 (次頁参照) のグラフ G1 のように表わされる。

(例2) 「グループ GR2 はエレメント EL4, EL5, EL6 からなる。プロセス PR2 は GR2 に対して更新モード・アクセスが許され、EL4 のみを更新することができる。」この記述は図-3 (次頁参照) のグラフ G2 のように表わされる。

DD/D は、このようなメタ・データのグラフをデータベースに編成したものである。ここでグラフ G0 は

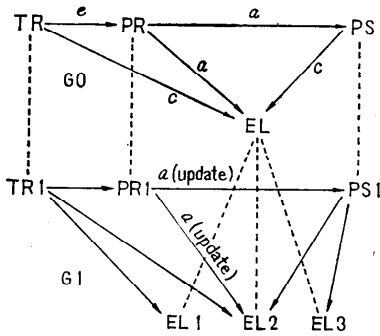


図-2 メタ・データ・グラフ G0-G1

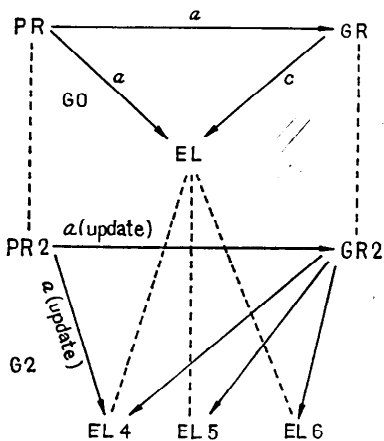


図-3 メタ・データ・グラフ G0-G2

DD/D におけるデータの型に、またグラフ G1, G2 などはデータの実現値に対応する。

実際の DD/D 編成として、'THREE D' SYSTEM の例を示そう。この DD/D の主要部は、以下の固有名称をもつ4種のデータベースから構成される。

- “データ・ディクショナリ”：DB または PR 名をエントリ・キーとして、PR, DB, PS, GR および EL の内容、構造、完全性などに関するメタ・データを蓄積する。
- “データ・ディレクトリ”：PS 名をエントリ・キーとして、PS, GR および EL のロケーション、物理特性などに関するメタ・データを蓄積する。
- “キーワード・データ”：エンド・ユーザ向けのキーワード表示子をエントリ・キーとして、“データ・ディクショナリ”内の対応する GR および EL へのインデクスとなる。
- “データ・エレメント・インデクス”：GR 名, EL 名あるいはそれらの別名をエントリ・キーと

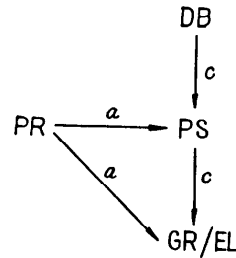
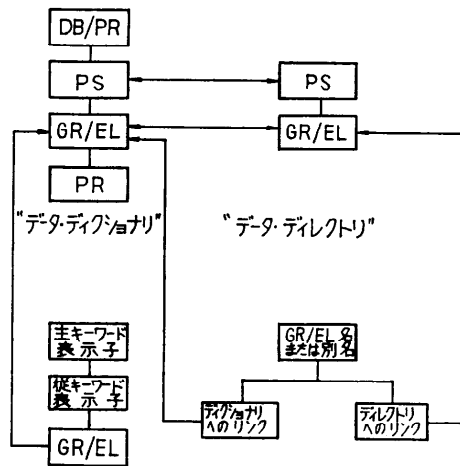


図-4 'THREE D' SYSTEM におけるメタ・データ・グラフ



“キーワード・データ” “データ・エレメント・インデクス”

図-5 'THREE D' SYSTEM の DD/D 編成図式

して、“データ・ディクショナリ” および “データ・ディレクトリ” 内の対応する GR および EL へのインデクスとなる。

図-4 はこのシステムにおけるメタ・データのグラフ表現を、また図-5 は DD/D 編成の図式を示したものである。

この DD/D では、PR, DB, PS, GR および EL について、次の属性記述が蓄積されている。

- PR：テキスト記述、使用する PS 名, GR 名 および EL 名, PR を起動するトランザクション名, 使用するサービス・ブロック, PR および PS の定義状態, PS の番地参照のためのテーブル名, PR, GR および EL を使うときのピクチャと編集規則。
- DB：構造 (PS および EL との関係), 装置の型とロケーション, アクセス法, 定義状態。
- PS：構造 (GR および EL との関係), 実現値

数の見積り値, 成長率, アクセス統計, 物理および論理関係, 定義状態.

- GR および EL: GR の構造 (EL との関係), テキスト記述, ピクチャ, 変位, 長さ, 編集規則, 安全性および完全性データ, キーワード表示子, GR および EL を使用する PR 名, 定義状態.

4. DD/D システムの機能

DD/D システムの機能目標には, ソースおよびオブジェクト定義の収集, 生成, 提供と, データの命名法, 使用法, 符号化などに関する標準化という2つの側面がある. 両者の重要度はシステム環境によって異なる. たとえば, 分散システムでは前者が重視され, 集中システムでは後者が強調されよう. また, 情報システムにおけるすべての要素間の関係が DD/D によって支配されるという考え方に立ち, 標準化されたメタ・データ管理を実施しようとするならば, すべてのメタ・データの登録, 改廃は DD/D に漏れなく反映されていなければならない. この前提のもとに, DD/D システムが本来もつべき機能をあげると, 次のように要約される.

(1) データに関する広範な定義づけ

データのもつさまざまな属性を明快に記述し, これを蓄積, 検索, あるいは更新する汎用的な手段が用意されていなければならない. つまり, メタ・データ記述言語と操作言語が必要である.

(2) データの矛盾性と冗長性の検知

メタ・データを受け入れる際に, それが DD/D に登録済みか, あるいは既登録のメタ・データと矛盾しないかを解析検知すること.

(3) メタ・データの版制御

同一の実体のメタ・データに, 旧版と新版, テスト版とプロダクション版のような, 複数版が存在する場合, その使用を制御すること. たとえば, 'THREE D' SYSTEM では, 提案, 同意, 承認, 有効および廃止という5種類の定義状態を区別している.

(4) 相互参照 (cross reference) 機能

DD/D に記述されている任意の要素間の関連性を維持すること. とくに, あるメタ・データが変更されたとき, 関連する他のメタ・データへの影響を検知し, 必要な変更手続きがとられること.

(5) 発見的検索機能

システムの開発, 運用段階で, 事前に予知できない計画や意思決定のために, メタ・データをその内容にもとづいて選択的に検索する問い合わせ応答機能.

(6) 標準ドキュメンテーション

ユーザ・クラスに応じて, 定義情報に関する標準ドキュメンテーションを生成すること.

(7) DBMS あるいは言語プロセッサに対するデータ定義ステートメントの自動生成

DBMS に対するデータ定義ステートメント, あるいは COBOL, PL/I などすべてのプログラム言語に対するソース・プログラムのデータ定義部を自動生成すること. またデータ定義変更の際は, 変更トランザクションを自動生成すること. この機能については, DBMS やオペレーティング・システムに対する適切なインタフェースを介して, オブジェクト定義を直接提供できることがのぞましい.

(8) 安全性と完全性の維持

メタ・データに対する不当な検索, 更新, 偶発的または故意の破壊から DD/D を保護すること. たとえば, DD/D の安全性モニタリング, 変更ロギング, 周期ダンプなどの機能.

実際に, 現行の DD/D システム・パッケージによって提供されている共通的な機能は以下のようなものである. 図-6 の機能構成と併せて参照されたい.

- ① DD/D (ただしデータ・ディクショナリが対象) の創成, 更新, および内容の維持.
- ② DBMS ディレクトリあるいはコピー・ライブラリの創成および更新用データ記述の生成.
- ③ DD/D への問い合わせに対する, データ属性,

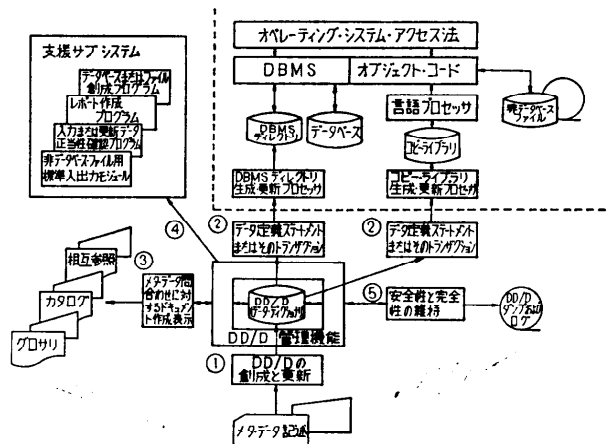


図-6 DD/D システムの機能構成

相互参照などのレポート作成または表示.

- ④ テストおよび支援サブシステムの生成
 - 非データベース・ファイル処理用標準入出力モジュール
 - 入力あるいは更新データに関する正当性確認プログラム
 - レポート作成プログラム
 - テスト用データベースあるいは非データベース・ファイルの創成
- ⑤ 安全性, 完全性, および性能維持
 - DD/D およびデータベースの安全性モニタリング
 - DD/D の変更ロギングと周期ダンプ
 - データベース利用統計の記録と解析

情報システムは DD/D によって駆動されるという考えを推し進めるならば, DD/D の情報にもとづいて自動的に制御ないし生成可能なものは, すべて DD/D システムの機能として具備されるべきだということになる. GUIDE の DD/D 要望では, データベース編成を制御するために, 次の一連のツールが提供されるべきであるとしている.

- データベース・システムの性能を測定し, 統計記録をとる, 測定ツール, これは汎用的なモデリングツールではなく, 特定 DBMS 専用の, 高精度のシステムでなければならない.
- 測定値を評価し, データベース・システムの性能改善のための変更の効果を予測する, アナライザまたはシミュレータ.
- DD/D の記述とアナライザの出力から, 性能向上をもたらす新しい記述を生成するシンセサイザ.
- シンセサイザによって生成された新しい構造に従って, データベースあるいはシステムを再編成し, その効果を測定する再編成ユーティリティ.

5. DD/D への要望と期待

DD/D は, オペレーティング・システム, DBMS, コミュニケーション・モニタ, 言語プロセッサ, ユーティリティなどの領域に深く関係する. メタ・データの集中制御という目的からすれば, システムのすべての成分は, DD/D を共通の定義情報源として利用できるようにすべきであろう.

しかし, 現実の DD/D システムは, DD/D の管理, システム成分とのインタフェースおよびユーザとのイ

ンタフェースに関して独自の機能を設けており, オペレーティング・システムや DBMS と機能構成が重複することが多い. とくに, DBMS との共存関係では, 図-7 に示すような3つの型がある.

(1) 完全分離型

DD/D システムは DBMS から分離独立している. DD/D は独自のアクセス法によって管理される.

例: LEXICON, DATAMANAGER

(2) 部分的結合型

DD/D システムは DBMS から独立しているが, DD/D のアクセス法として共存する DBMS を併用する. つまり, DD/D システムは DBMS のアプリケーション・サブシステムとして実現される.

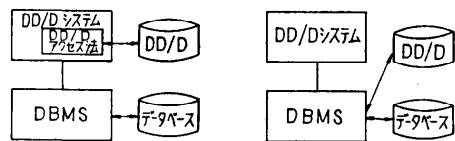
例: UCC TEN, 'THREE D' SYSTEM

(3) 完全統合型

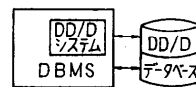
DD/D システムは DBMS に統合される. GUIDE の DD/D 要望, および次に述べる ANSI/X3/SPARC のデータベース・システム・モデルは, この型の DD/D を志向している.

(1)または(2)の型の DD/D システムは, ディクショナリ管理を主体とし, ディレクトリ管理は DBMS に依存する. ディクショナリとディレクトリを2つの異なる管理系に分離することは, メタ・データの冗長性と矛盾性をもたらす原因となる. さらに, メタ・データ参照の重複による性能低下を招くことにもなりかねない.

ANSI/X3/SPARC (the Standards Planning and Requirements Committee) のデータベース・システム調査グループは, DD/D を核とするデータベース・システム・アーキテクチャについての提案²⁾をまとめた. この提案は, DBMS 標準化に関する判断基準を追求する目的で, システムを構成する成分と成分間の



完全分離型 部分的結合型



完全統合型

図-7 DD/D システムと DBMS の関係

インタフェースを識別し、それらの機能のあり方について論じたものである。

SPARC のデータベース・システムは、次の3つのモデルと、それぞれに対応するスキーマからなる。

- 概念モデル：データベースに反映しようとする実体について、企業基準、契約、法律などにもとづく普遍的な情報を維持しようとするものである。その属性記述が概念スキーマであり、他のモデルを律する基準として存在する。概念スキーマはデータベースに対してただひとつだけ存在し、企業管理者 (enterprise administrator) によって管理される。
- 外部モデル：個々のユーザあるいはアプリケーション・プログラムから見た実体を表現する対象の集りであり、アプリケーション・プログラムからの要求に応じて具現化される。実体の属性はユーザ・クラスに対応して外部スキーマに記述され、複数のアプリケーション・システム管理者によって管理される。
- 内部モデル：物理媒体上に蓄積されるデータの集りとして実在するもので、その属性は内部スキーマに記述され、データベース管理者によって管理される。

概念スキーマはもっとも安定したメタ・データ源である。外部および内部モデルに変更があっても、外部スキーマと概念スキーマ間、および概念スキーマと内部スキーマ間の写像規則を変更するだけで、高度のデータ独立性が達成される。DD/D は3つのスキーマおよびスキーマ間の写像規則を蓄積したデータベースで

ある。図-8 に示されるように、SPARC のモデルでは、システムの主要成分と DD/D の間のインタフェースを定義することに努力が払われている。さらに、データ・モデルに関するスキーマ以外に、システム・コントロール、プログラムの準備、ジョブ管理、資源管理およびメッセージ管理に関するメタ・データも、DD/D に統合されるか、あるいは DD/D に関係づけられる必要がある。

ところで、メタ・データ管理をデータ・セマンティクスの領域にまで拡張しようとするとき、DD/D は限界に直面する。DD/D では、データあるいはデータ間関係のもつ意味が単なるテキストとして記述されているに過ぎず、システムが意味論的な解釈と操作を施すわけにいかないからである。データ・セマンティクスの完全性を維持するためには、セマンティクスを構造的に表現し、蓄積する手段が必要である。

この問題に対するひとつのアプローチとして、トロント大学 CSRG (Computer Systems Research Group) によるセマンティック・ネットワークの研究がある。

このネットワークは、ラベル付きの節とラベル付きの枝からなる有方向グラフとして表現される。節の型には、概念、事象、特性および値がある。概念節は、モデル化しようとする世界の本質的な定数やパラメータであって、物理的あるいは抽象的な対象を指定するものである。事象節はモデルにおけるアクションを表わすもので、その意味づけは、事象節および事象について起りうるケースを示すいくつかの節と枝によって表現される。たとえば、図-9 (次頁参照) のセマンティック・ネットワークにおいて、事象節 'SUPPLY' の意味は、'agent' および 'source' の役割をもつ 'SUPPLIER'、'destination' の役割をもつ 'PROJECT'、および 'object' の役割をもつ 'PART' によって表わされる。特性節は、概念、事象、あるいは他の特性の状態を表現したり、修飾したりするのに用いられる。グラフ上では、特性節から修飾の対象となる節への枝 'characterize' および特性節から対応する値節への枝 'value' によって表現される。たとえば図-9 で、特性節 'DATE' は、値節 'DATE・VALUE' によって、事象節 'SUPPLY' を修飾している。値節は特性の値を表わすためのものである。

データあるいはデータの構造は、システムに定義されたセマンティック・ネットワーク

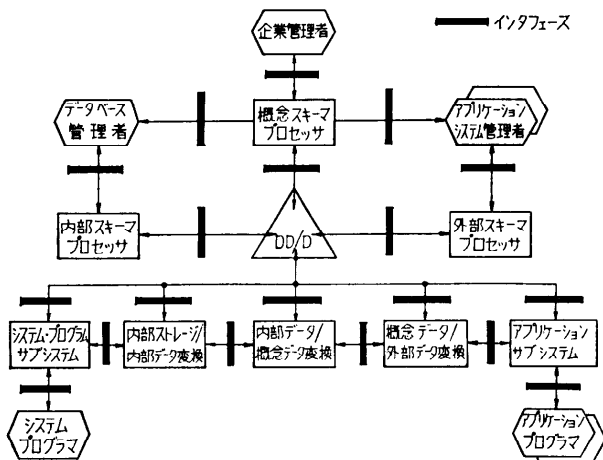


図-8 SPARC データベース・システム・アーキテクチャ

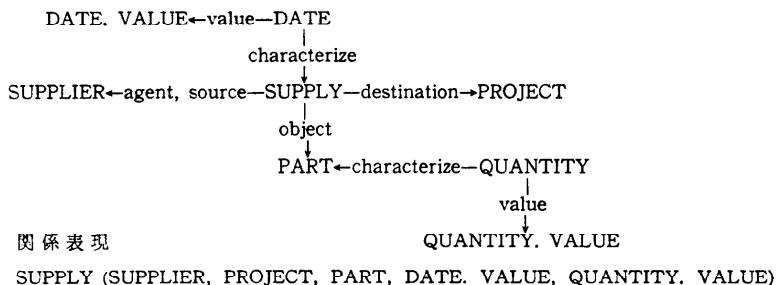


図-9 セマンティック・ネット・ワーク

に適合するときのみ、システムにとって意味あるものとして受け入れられる。データベースのセマンティクスを蓄積するために、セマンティック・ネットワークは関係 (relation) として表現される。図-9 の関係 SUPPLY はその例である。そして、セマンティック・ネットワーク上に新しい節を生成するための操作は、対応する関係上の操作として考察されるわけである。

データ・セマンティクスの研究は、データ・ベースシステムの知能化を促進する手段のひとつとして、今後の発展が期待される。

6. おわりに

DD/D は、最近の分散システムの発展にともなって、システムの構造と働きを制御する情報源として、その支配領域を拡げつつある。データベース・システムに限らず、情報システム全般を DD/D の立場から見直すことはきわめて意義深いことと思われる。

参考文献

- 1) GUIDE DD/D Project : Data Dictionary/Directory Requirements, GUIDE International Corporation (1975).
- 2) ANSI/X3/SPARC : Study Group on Data Base Management Systems Interim Report, 75-02-08, FDT Bulletin of ACM-SIGMOD Vol. 7, No. 2 (1975).
- 3) データベース専門委員会：データベース・システムに関する調査—データ・ディクショナリ/ディレクトリ、日本電子工業振興協会 (1976).
- 4) P. P. Uhrowczik : Data Dictionary/Directories, IBM SYSTEM JOURNAL Vol. 12, No. 4 (1973).
- 5) N. Roussopoulos and J. Mylopoulos : Using Semantic Networks for Data Base Management, First International Conference on Very Large Data Bases (1975).

(昭和51年11月4日受付)
(昭和51年12月19日再受付)