

データディクショナリ管理システム

斉藤 昇 (三菱電機(株)情報システム部 関東機械計算室)

1. はじめに

DDMS (Data Dictionary Management System) は当社開発のDD/DS (データ・ディクショナリ・ディレクトリ・システム) である。近年DD/DSの導入を検討する企業が増えていると言われている。⁽²⁾ アメリカでは1970年代の前半から市販のDD/DSが登場し導入が始まっているのに比べ、⁽¹⁾ 我が国では、導入の気運は高まってきているものの、導入例はまだ極めて少ない。⁽²⁾⁽³⁾ このような中で、当社は82年にDDMSの自社開発を行い導入を開始しているため、その背景やシステムの内容、今後の展開について述べる。

2. DDMS開発の背景とねらい

近年、企業内システム部門の生産性向上が極めて重要な課題となっている。当社においても、情報システム開発・保守の生産性向上をねらいとして、81年より5年計画で各種の施策が推進されている。既に、システム開発標準の制定、会話型プログラム開発環境の整備、開発支援ソフトウェア・ツールの開発(プログラム・ライブラリ管理ツールやテスト・データ生成支援ツール)、パターン別の標準プログラムの整備などが実施されており、DDMSもこの活動の一環として、次のねらいで開発・導入されたものである。

(1) システム開発・保守業務支援

情報システムの範囲と規模の拡大、およびシステムの広域化が急速に進んだ結果、当社の場合、保有プログラム量は約2900万ステップに達し、しかも毎年の増加は約600万ステップという状況であり(82年度の実績値)、人手による管理ではもはや追いつかなくなっている。しかも、業務内容を見ると、既存システムの改善や保守、作り替え作業の比重が増えているほか、新規業務の導入であっても、他の既存システムのメンテナンス作業に波及することが多い。このようなシステム環境の変化に応じて、プログラムを変更し、他システムとのインターフェースの整合性をとることは、負荷的にも技術的にも容易ではなくなっている。そのため、システムの理解不足によるトラブルを頻発したり、安易なその場しのぎの処置による改定作業の結果、システムはますます複雑になるという悪循環を引き起こしている。このような問題を解決するためには、情報システムがどう構成されているか、とくにプログラムと、ファイル、インプット、アウトプット、データ項目とがどういう関係になっているかを管理することが大切であり、DDMSはそのための管理手段を提供する。

(2) メタデータの標準化・一元化

メタデータとは、一般には、データの構造や属性をデータ定義言語(DDL: Data Description Language)で記述したものをさすが、ここではもっと広い意味で、ファイルやインプット、アウトプット、データ項目に関する自然言語での記述(通常の手書のドキュメント)も含めて考える。メタデータのうち、とくにデータ項目については、同一のデータ項目が複数の名前を与えられて別々に管理されていたり、逆に、異なるものと同じ名前をつけてしまうといった混乱が多い。従って、システムの関係者どうしのコミュニケーションを、正しく、円滑に、効率良く行うためには、単に情報システムの構成を管理するだけでは不十分であり、メタデータの標準化・一元化がぜひとも必要である。DDMSにより、単一のデータ・ディクショナリでメタデータを管理することで、このことが実現できる。

(3) システム開発支援ツールの統合化

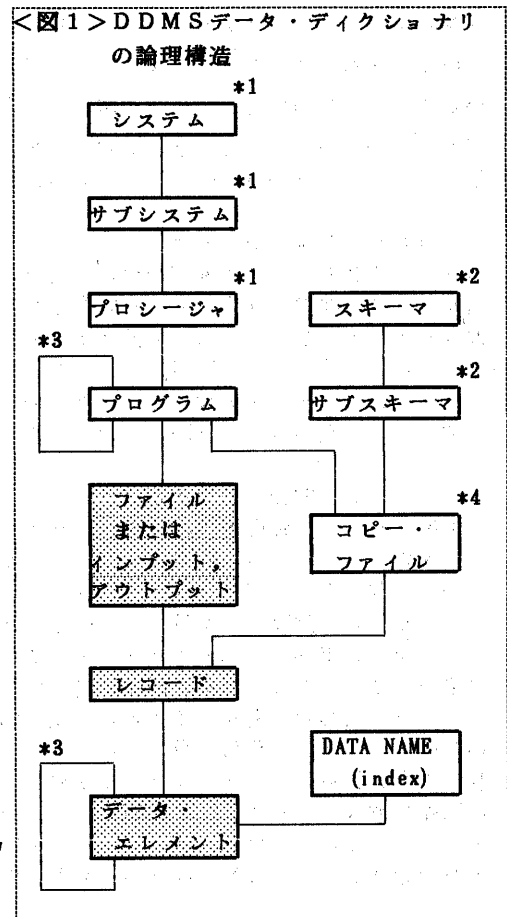
ディクショナリに蓄える情報をシステム開発段階から活用するためには、また、ディクショナリの内容の質を保証するためには、ディクショナリへの情報登録を、開発段階から情報発生 の都度行うべきである。そのためには、開発を急ぐあまり登録作業は後回しにならないように、システム開発標準の中できちんと登録作業を位置付け、徹底する必要がある。さらに、各種の支援ツールをディクショナリを核として統合化し、ツール全体としての総合的かつ強力な開発支援機能をもたせることが重要である。このような観点からDDMSは、テスト・データ生成、プログラム生成、ドキュメント作成などの開発支援ツールと統合化してゆくよう拡大を図っている。

3. データ・ディクショナリの内容

図1にDDMSが管理するディクショナリの論理構造を示し、表1にディクショナリ登録情報の内容を示す。ディクショナリに入れるべき情報はいろいろ考えられるが(例えば文献(6)参照)、ディクショナリを正しく維持するためには、何に利用するのかねらいを明確にして内容を決めるべきである。当社の場合、将来的にはドキュメントを含めた情報システムの全情報をディクショナリをベースとして管理する予定であるが、当面は次の2点にねらいをしばっている。

- 情報システムの構成を管理し、変更の影響度分析などを行い、開発・保守効率の向上を図る。
- メタデータを標準化するとともに、その内容をシステム開発段階にも活用する(図1の網かけ部がこれに対応した情報である)。

相互関係としては図1の線で結ばれたものどうしの関係を持っており、全て両方向(即ち、使っているものと使われている先)の関係が付けられるようになっている。また、プログラム間(サブプログラム CALL)の関係や、データ・エレメント間(集団項目とその子項目との関係)も定義できる。さらに、データ・エレメントについては、プログラム・コーディング上の DATA NAME をインデックスとして付けられるようになっているため、ディクショナリ登録キーだけでなく DATA NAME でのアクセスも可能である。



- *1…当社のシステム開発標準では情報システム構造をこの3階層に分けてとらえる。簡単に言えば、システムは開発対象業務全体を指し、サブシステムは1つのまとまった業務機能単位を指し、プロシージャは連続して遂行される処理のまとまり(コンピュータ処理では、ほぼジョブに相当)を指す。
- *2…CODASYL型データベースで言うスキーマ、サブスキーマのことである。
- *3…それぞれ、プログラム間の関係、データ・エレメント間(集団項目とその子項目との関係)の関係を表した線である。
- *4…COBOLプログラム上のデータ定義情報などを外部ファイル化したものであり、いわゆるコピー・ライブラリの1つのメンバーに相当する。

<表1> データ・ディクショナリへの登録情報の内容

登録要素	登録内容
システム サブシステム プロシージャ インプット アウトプット スキーマ サブスキーマ コピーファイル	登録キー (*1) 名称
プログラム	登録キー 名称 オブジェクト名 <使用ファイル情報(*2)> 生成/更新/検索の区分 ファイル・アクセス方法
ファイル	登録キー 名称 ファイル編成 ブロッキング 備考
レコード	登録キー 名称 レコード長 改定記録(*3) <構成データ・エレメント情報(*2)> 構成データ・エレメント名 (DATA NAME) レコード・キー情報 データタイプ・桁数(*4)
データ・エレメント	登録キー 名称 (カナ名称と DATA NAME) 備考 データタイプ・桁数(*4)

- *1…ディクショナリへ登録するものには一意に識別できるようなキーを付ける。実体は今まで運用していたシステム・コードやプログラム番号などであるが新しく設けたものもある。
- *2…これらについては要素間の相互関係に付随して情報を持てるようになっている。
- *3…当ディクショナリは変更履歴を持たないが、レコードについては改定内容のコメントを入れられる。
- *4…データ・タイプや桁数はデータ・エレメントの方にはファイル蓄積時のものを登録する。インプットやアウトプットの中には、同じデータ・エレメントであっても、タイプ・桁数がこれと異なる場合があるので、その場合はタイプ・桁数をレコードとの関係の付属情報として登録する。こうすることにより、同じデータ・エレメントは1個だけディクショナリに登録するだけで済む。

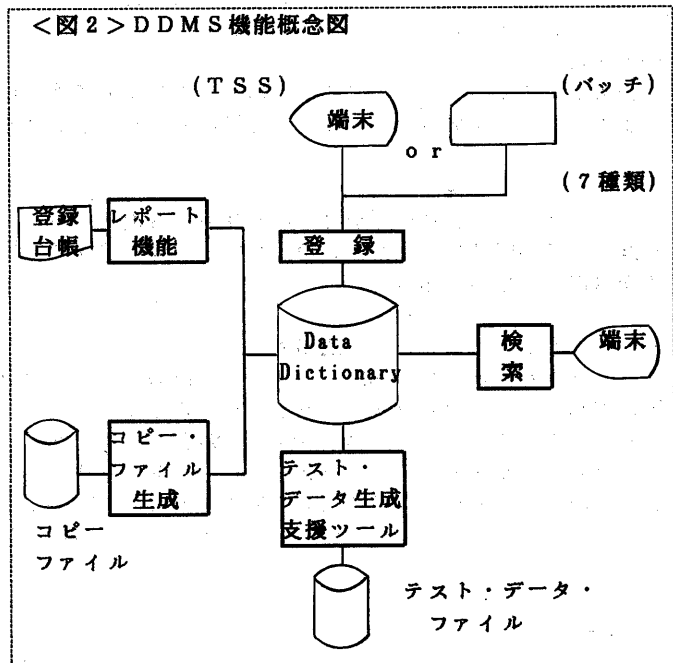
4. DDMSの機能

4.1 登録機能

ディクショナリへのデータ登録は全て「定義書」と称する7種類の様式に従って、バッチまたはTSSでインプットする。TSSの場合は即時にディクショナリを更新する。

ソース・プログラムやJCLなど、ある程度自動的に相互関係をつかめる情報はあがるが、その内容は登録要素を一意に特定できるほど標準化されていないため、自動登録機能は持たせていない。従って、現在は全てインプットしなければならないが、開発中のシステム設計CADツールから自動的に登録されるようにする予定である(6.今後の展開の項参照)。

<図2> DDMS機能概念図



4. 2 検索機能

構成や使用先など全部で16パターンの検索ができる。ここで、構成とは指定した要素が「使っているもの」の検索であり、使用先とは指定した要素が「使われている先」（いわゆる“where-used”）の検索である。以下にプログラムの場合の例を示す。

●プログラム構成検索の検索対象

- ・使用サブプログラム（指定したプログラムがCALLしているサブプログラムを検索する。孫以下の検索も可）
- ・使用ファイル
- ・使用コピー・ファイル

●プログラム使用先検索の検索対象

- ・使用先プログラム（指定したプログラムをCALLしているプログラムを検索する。さらにその親以上の検索も可）
- ・使用先プロシージャ（指定したプログラムが使われているジョブ）
- ・使用先サブシステムおよびシステム

検索はTSSで行うことが主であるが、バッチで実行することも可能である。検索の機能により、変更の影響度分析が以前より正確かつ迅速にできるようになる。

デクショナリで相互関係付けられている全ての関係の検索が可能であるが、当面実用的なパターンのみ検索プログラムを作成した。例えば、プログラムとデータ・エレメントの関係などを把握できれば便利であるが、このデクショナリで分かるのはファイルとレコードを介した関係であって、プログラムがアクセスしているデータ・エレメントの関係ではない。後者の意味での関係を付けることは現状ではほとんど不可能であり、したがって、検索パターンも用意していない。

4. 3 テスト・データ生成支援ツールとのインターフェース

テスト・データ生成支援ツールは、DDMSとほぼ同時に自社開発したツールであり、次のようなデクショナリ・インターフェースを持っている。

●項目指定をデクショナリに登録してあるデータ・エレメント名で行えるため、従来のカラム指定のやり方に比べ分かりやすい。

●ファイル内容をリストするとき、項目毎に編集され、しかもデータ・エレメント名がタイトルとして表示されるので、ファイル・レイアウトを見なくとも内容の確認ができ、作業効率が向上する。

将来は、デクショナリにデータのとりうる値の条件を登録し、この条件にもとづいてデータ値を発生する機能を持たせるなどの拡張を行う予定である。

4. 4 レポート機能

リクエスト・バッチにて登録台帳を作成する機能である。リクエストは登録要素の種類毎に、登録キー値の範囲指定で行う。登録台帳の主な用途は登録状況管理用であり、デクショナリ管理者向けのものである。

4. 5 コピー・ファイル生成機能

レコードを指定して、COBOL プログラム用のレコード定義部分（これをコピー・ファイルと呼んでおり、いわゆるコピー・ライブラリの1つのメンバーのことである）を生成する。中間ファイルの場合など、1つのプログラムが同一レイアウトのファイルを複数使う場合があるので、コピー・ファイル内のデータ・エレメント名には、デクショナリ登録のデータ・エレメント名の前に prefix を付けた形で生成することができる。したがって、1つのレコード定義から、prefix を分けて複数のコピー・ファイルを作ることができる。

5. DDMSの実現方式

5.1 データ・ディクショナリの構造

DDMSのディクショナリは当社のCODASYL型DBMSであるEDMS (Extended Data Management System) で実現されている。そのデータベース構造を図3に示す。

(1) 主体レコード

登録要素対応の情報を持つレコードである。主体の種類(システム、サブシステム、・・・、データ・エレメント)を示す区分と、主体の種類に応じた情報(表1の登録内容のうち、備考、相互関係に付随した情報以外の情報)を持つ。

(2) 関係レコードと構成セット、使用先セット

主体間の相互関係を持つためのレコードとセットである。図1の登録要素間を結ぶ線に相当するセットはそれぞれ、構成、使用先の関係に対応している。

(3) 主体付属情報レコード

ファイルやデータ・エレメントの備考、レコードの改定記録を持つレコードである。主体に持ってもよい情報であるが、スペース効率を上げるためレコードを分けている。

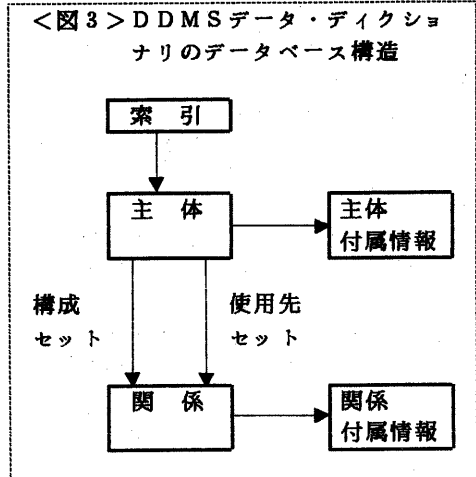
(4) 関係付属情報レコード

相互関係に付随した情報を持つレコードである。やはりスペース効率を上げるため、関係レコードとレコードを分けている。

(5) 索引レコード

DATA NAME からデータ・エレメントを検索するための索引のレコードである。

以上のように構造が汎用化されているため、登録内容の拡張が容易にできるデータベースとなっている。



5.2 DDMSのソフトウェア構成

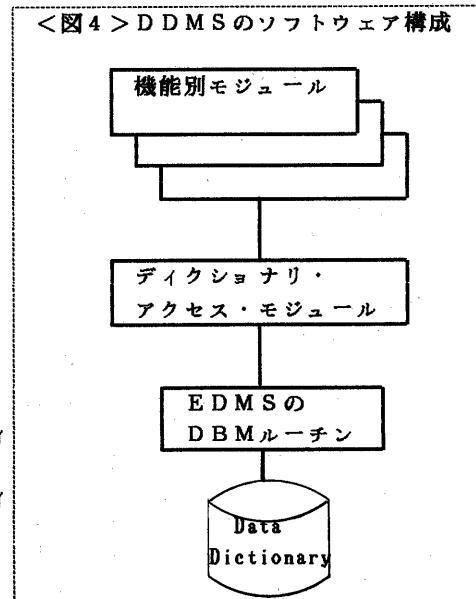
図4にDDMSのソフトウェア構成を示す。

(1) 機能別モジュール

4項で説明した各機能を実現するモジュールである。ディクショナリのアクセスは、EDMSのDBMLルーチンを使うのではなく、ディクショナリ・アクセス・モジュールを介して行う。

(2) ディクショナリ・アクセス・モジュール

DDMSディクショナリ・アクセス専用のサブルーチン群である。このモジュールの役割は、DBMLルーチンより高度なディクショナリ・アクセス機能を提供して機能別モジュールの負担を軽くすることである(構成や使用先関係を順にたどる機能、使用先や索引のリンクを自動的に付ける機能、ディクショナリ共用時の排他制御機能などがある)。また、いわゆるDBMS独立が達成されており、ディクショナリをEDMS以外のファイル形態に移行することができる。このモジュールの入れ替えだけで簡単に行える。



6. 今後の展開

当社は、次のステップとして、システム設計CADツールを開発中である。これはシステム設計段階で作成される、業務処理流れ図、システム・フローチャート、I/O 帳票レイアウト、画面レイアウト、ファイル・レイアウト、ランチャート（ジョブステップの流れ図）などの図で表現されるドキュメントの作成を支援するCADシステムである。図5にその構成とDDMSとの関係を示す。（網かけ部が開発中のシステム設計CADツールである。）

(1) 会話型ドキュメント作成

上記のドキュメントをグラフィック端末で作成する。DDMSと次のインターフェースを持つ。

●作成したドキュメントに含まれるディクショナリ情報を自動的に登録する（ディクショナリへのインプット作業の自動化）

●ディクショナリ登録済の情報について、ディクショナリを参照し、ドキュメントの中にその内容を入れる（ディクショナリ登録情報の活用）

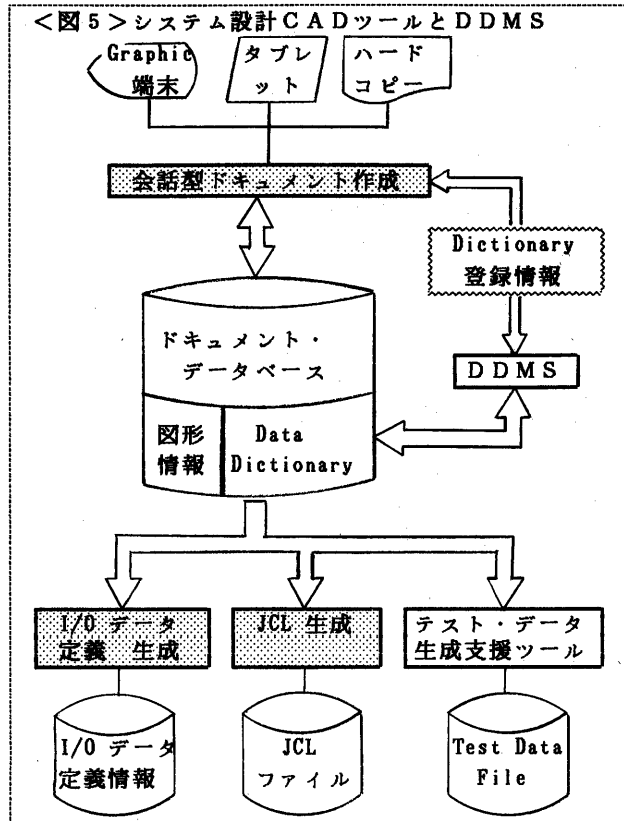
(2) I/O データ定義生成

プログラムに必要なI/O データ定義や、オンライン・システム用の画面定義情報を、I/O レイアウトや画面レイアウトの情報から生成する。

(3) JCL 生成

ランチャート情報からジョブ・コントロール・コマンド・ファイルを生成する。

システム設計CADツールにより、ディクショナリへのデータ登録は、DDMSの登録機能によるインプットが不要になり、ドキュメント作成作業と一体化して自動的に行われるようになる。今後開発するツールも、このようにDDMSとのインターフェースをとり、ツールの統合化を図ってゆく予定である。



<参考文献>

- [1] 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— (データベース専門委員会報告書) 昭和51年3月 日本電子工業振興協会
- [2] データベースの高度利用を目指すデータ・ディクショナリ活用技術を探る 斉野 日経コンピュータ 1982.5.17
- [3] 導入・稼働に2年はかかるデータ・ディクショナリ 斉野 同上 1983.10.31
- [4] データ・ディクショナリ/ディレクトリ (DD/D) ・システムの現状と将来 —広島大学における実例— 池田 同上 1982.12.27
- [5] データ・ディクショナリの導入と設置 竹中 EDP リサーチ・レポート 1978.4.1
- [6] 特集 情報資源管理の考え方とDDS 小柳 他 Computer Report 1981.5
- [7] データ・ディクショナリ/ディレクトリ 椿 情報処理 1982.10