

---

---

# マルチプラットフォームを統合管理する

## データ辞書システムの構築

エムジーシーコンピュータサービス㈱

---

### 執筆者 Profile



関 口 尚 樹

1997年 エムジーシーコンピュータサービス入社  
1997年 基幹系システム保守を担当  
1999年 有価証券システム開発を担当  
1999年 データ分析ツール開発を担当  
2000年 システム再構築開発を担当  
現在，再構築チーム所属



上 田 淳 一 郎

1999年 エムジーシーコンピュータサービス入社  
1999年 有価証券システム開発を担当  
1999年 データ分析業務を担当  
2000年 物流システムを担当  
現在，再構築チーム所属



小 笹 ひ ろ み

1997年 エムジーシーコンピュータサービス入社  
1997年 データ分析業務を担当  
現在，再構築チーム所属



岩田夏子

1998年 エムシーシーコンピュータサービス入社  
1999年 有価証券システム開発を担当  
1999年 データ分析ツール開発を担当  
2000年 システム再構築開発を担当  
現在、再構築チーム所属

## 論文要旨

当社が進めているシステム再構築や新規システムの追加開発では、既存システムと新システムの連携が必須である。しかし、既存システムと新システムの項目リンクが把握できないまま新システムを導入すると、複数ファイルで重複項目が作られてしまい、運用や保守の負担が増大する。

そこで、各システムが保有しているデータ項目情報を統合管理でき、それをシステム開発者とエンドユーザの両者が有効に活用できる「データ辞書システム」を構築することにした。

これは「統一項目」、「コードドメイン」という概念によって、データ項目情報と一般的な業務用語及び他システムで使われている用語をリンクづけて統合管理するものである。

本システムは、システム管理者に対して統合的な管理機能を提供する。また、エンドユーザに対しては、検索機能を開放することで公開されたデータの二次利用を可能としており、情報収集・分析に有効利用されている。

今後、データ項目情報の対象範囲・質を拡充すると同時に、Web化によって利用ユーザ数を拡大し、さらなる情報資源の統合化を推進していく。

## 論文目次

<b>1 . はじめに</b> .....	《 5》
<b>2 . 背景</b> .....	《 5》
2 . 1  現行システムの問題点	
2 . 2  システム再構築の方向性	
2 . 3  データ辞書システムの必要性	
<b>3 . 開発の経緯</b> .....	《 7》
3 . 1  要件	
3 . 2  データ辞書システム概念	
3 . 3  ハードウェア及びソフトウェア構成	
3 . 4  構築の規模	
3 . 5  利用者側の処理	
<b>4 . 主な工夫点</b> .....	《 14》
4 . 1  規約による標準化	
4 . 2  検索機能の充実	
4 . 3  ヒアリング内容のデータベース化	
<b>5 . 評価</b> .....	《 14》
5 . 1  エンドユーザの評価	
5 . 2  システム開発者の評価	
<b>6 . 今後の展望</b> .....	《 15》
6 . 1  分社化及び他企業間インターフェース増加への対応	
6 . 2  Web 化とXML展開	
<b>7 . むすび</b> .....	《 16》

## 図表一覧

図 1	インターフェースの増加	《 6》
図 2	データベース辞書の概念図	《 7》
図 3	ファイル設計概念図	《 8》
図 4	統一項目の考え方	《 8》
図 5	コードドメインの考え方	《 9》
図 6	新システム構成イメージ図	《 10》
図 7	アプリケーションの範囲	《 11》
図 8	データ辞書システムの利用例 1	《 12》
図 9	データ辞書システムの利用例 2	《 13》
表 1	データ辞書への登録状況	《 11》

## 1. はじめに

当社の親会社は「独自技術によるユニークな製品の提供」を会社理念として、基礎化学品などの大型バルク製品からプリント配線基板、高機能樹脂製品などに代表される機能材製品まで幅広い事業領域を展開する中規模総合化学会社である。

当社は、親会社のシステム部から平成6年に分離独立した企業で、富士通の大型ホストコンピュータによる営業・資材・経理などの業務システムの運用・保守や、ホストコンピュータ・クライアント/サーバ連携システムの企画・開発などを担当している。

## 2. 背景

### **2.1 現行システムの問題点**

保守を担当している基幹系システムは1970年代に大型ホストコンピュータを中心に構築され稼動していたが、以下のような問題が発生していた。

(1) データベースの分散

事業拡大のつど個別にシステム開発が行われてきたため、事業所単位でシステムが構成されており、データベースが分散化している。

(2) データの共有性や統合性に欠ける

事業所ごとの要望に従い設計を行ってきたため、他事業所システム間との整合性がとれておらず、データの共有性や統合性に欠けるシステムになってしまっている。このため保守作業や他システムとの連携処理が必要なプログラム開発を行うごとに、大きな労力がかかっていた。

(3) メンテナンス負荷の増大

現行システムでは、業務の流れにあわせて同じデータ項目を複数ファイルに持つ、正規化されていない「業務中心型システム」が残されている。このため、新しい切り口のプログラムを開発するごとにファイルが増え、メンテナンスの負荷が増大していた。また、複数ファイルに重複項目を持つため、一個所データを修正するとほかのファイルの同項目にも修正を反映させなければならず整合性を保つのが困難であった。(図1)

この悪循環を断ち切るためにはファイル設計を業務中心型からデータ中心型に切り替える必要がある。



状態が把握しづらい。これを解決するには、データ項目情報を整理し公開する必要がある。さらに、ERPパッケージで管理できない部分に関しては、別パッケージ導入または自社開発を行うことも考えられ、マルチプラットフォーム化に対応した、より複雑なデータ項目管理が必要になる。

そこで、システム再構築を行うにあたり各パッケージソフト導入と並んで、既存システムとのリンク情報を管理する「データ辞書システム」を導入することとなった。

この「データ辞書システム」が実現すれば、インターフェースを新規に作り込む際の辞書として活用し、無駄な重複項目のないファイル設計を行うことが可能になる。また、将来的に複数のパッケージを導入することになった場合でも、分散する項目名の管理を行えるため、インターフェースとなる新ファイルの作成を抑え、システム連携をはかる上での支援ツールになる。このことによって、新システムを将来的に拡張することも可能になった。(図2)

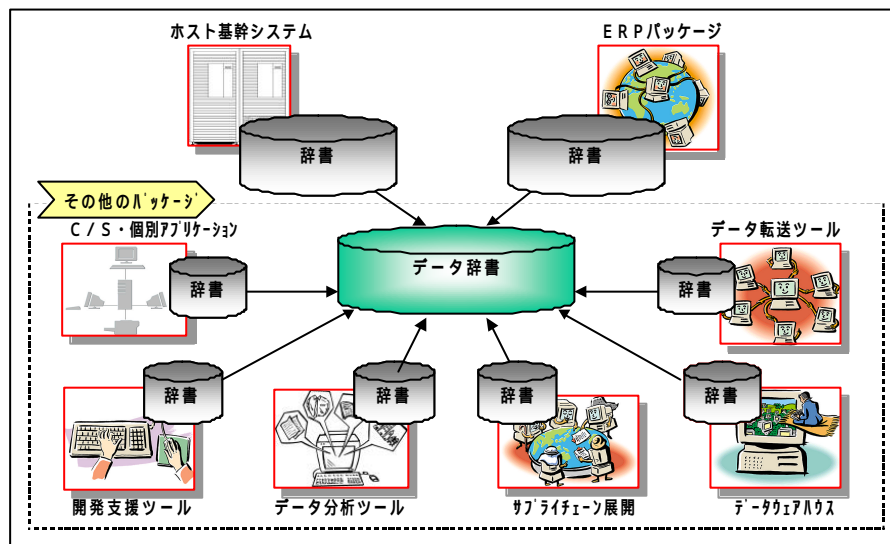


図2 データベース辞書の概念図

### 3. 開発の経緯

#### 3.1 要件

データ辞書システムを開発するにあたり、すでに市販されているソフトウェアで以下の条件を満たすものがないか調査を行った。

(1) 3つの辞書機能

- (a) 異なるシステム上に存在する、正規化により統合されるべき「同義語」のデータ項目情報を提供する「同義語辞書」
- (b) プログラムロジック内で管理しているコード体系に関しても、コード体系を参照することのできる「コードブック辞書」
- (c) あるデータ項目がシステムのどのファイル（または画面・帳票）にあるのかの情報を提供する「影響範囲特定辞書」

- (2) 属性・桁数が一致しない場合でも「同義語」を管理できること
- (3) 各種辞書機能を簡単に検索できる機能

調査を行なったが、要件を満たすものが存在しなかったため自社開発を行うことにした。

### 3.2 データ辞書システムのご概念

データ辞書システムのファイル設計概念図を以下に記す。

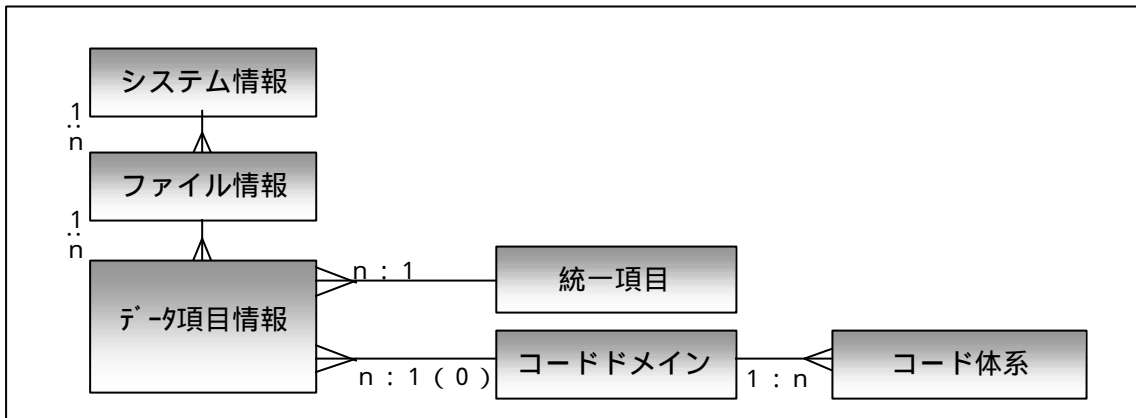


図3 ファイル設計概念図

各システムにはn個のファイルがあり、その1ファイルにはn個のデータ項目が存在する。この関係をそのままファイル設計に生かし、上記のような設計を行った。

また、この最小単位である「データ項目」に対して「統一項目」「コードドメイン」を付加することによって、他システム間とのデータリンクを容易に実現することができる。

以下にその概念を述べる。

#### <統一項目>

図のように、データ項目名は業務で使い慣れた名称をそのまま用い、その代わりにまったく同じ内容・意味のデータ項目をグループ管理するために命名したものが「統一項目」である。統一項目の考え方を図4に示す。

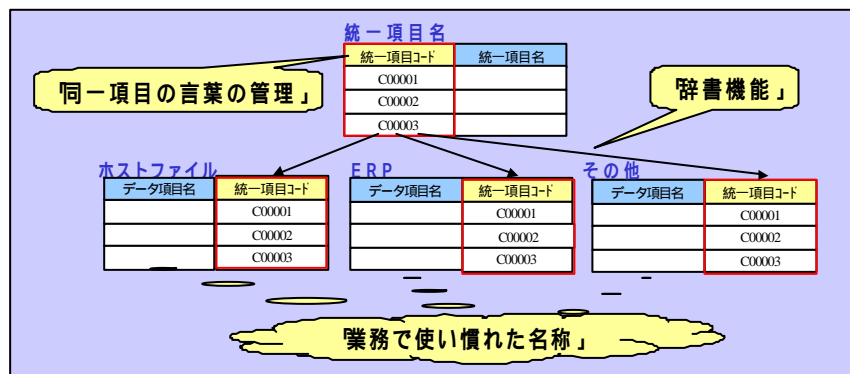


図4 統一項目の考え方



統一項目は、属性・桁数にとらわれず管理を行う。この仕組みによって、システムごとに桁数は違うが同じ内容がセットされるデータ項目も、同一グループとして管理可能になる。

統一項目を使用することにより、既存ファイルを正規化せずと同じ働きを持つデータ項目をグループ管理できるようになる。このため、システム間の連携処理を開発する場合でも統一項目に着目してファイル設計をすれば、作業工数をかけずにデータ項目の無駄な増大を抑えることができる。また、ユーザは新データ項目を覚えていなくても、データ辞書を調べれば確認できるため、業務に支障をきたさない。このことが、システム再構築でユーザから高く評価された。

### <コードドメイン>

コード体系が等しいデータ項目をグルーピングして命名したものがコードドメインである。コードドメインの考え方を図5に示す。

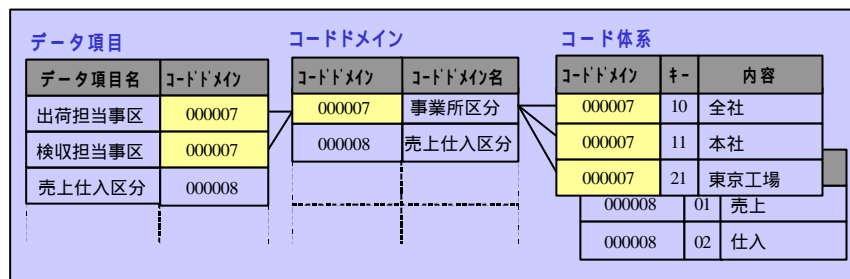


図5 コードドメインの考え方

コードドメインは、データ項目とコード体系の属性・桁数を一致するものの橋渡し役を担い、これらをリンクさせることにより、整合化されたシステムコード管理を実現することができる。またコードドメインを管理することにより、各々のシステム開発において同一の属性を持つデータ項目やコード体系が安易に増殖することを防止でき、効率的なデータ項目管理を実現することができる。

以上がデータ辞書システムの主な概念である。

データ辞書システムはこの「統一項目」と「コードドメイン」という概念によるデータリンク機能を中心に据え、すべての情報をシステム単位・ファイル単位・データ項目単位ごとに一元管理する点を最大の特徴として設計した。

すべての情報を一元管理することによって、必要なファイルやデータ項目の所在を明らかにし、重複する無駄な新ファイル・新データ項目の増加を防ぐことができる。さらに、統一項目やコードドメインも一元管理されているため、正規化した際の情報を管理することができる。

また、当社の開発者は、使用するファイルのレイアウトをドキュメント管理者から入手して開発を行っている。このとき、最新のファイルレイアウトを設計者個人で保管している場合があり、ファイリングされたドキュメントが古いために混乱が生じることがあった。このような情報をすべて一元管理することによって、開発者は自分で簡単に最新ファイルレイアウト情報を出力することが可能になる。

### 3.3 ハードウェア及びソフトウェア構成

新システム構成イメージ図を以下に示す。

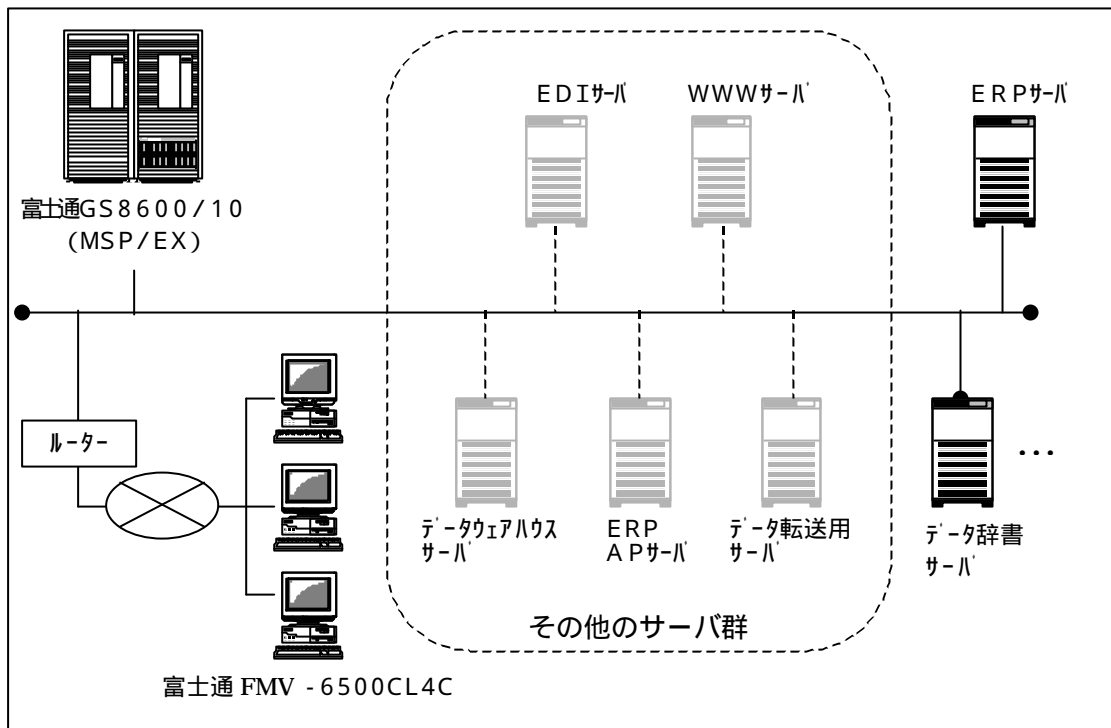


図6 新システム構成イメージ図

既存のホストコンピュータである富士通GS8600/10とERPサーバが連結されており、データ辞書サーバはその間に配置している。このデータ辞書サーバが複数システムがもつデータ項目情報の一括管理を行う。

ERPパッケージを導入するにはクライアント環境に高性能が求められる。そこで、クライアント機にCPU Celeron500、メモリ128MBと高性能のFMV6500CL4Cを数百台導入して開発者及びエンドユーザに配布する計画であり、今後はさらにFMVの上位機種購入も検討している。

辞書機能として当初はイントラネットを介したWeb化を考えていたが、工数短縮のためVisualBasic6.0のアプリケーションとなっている。データベースは、リレーショナルデータベースの特徴を生かせるシステムを構築する必要があったため、Oracle8iを採用した。

### 3.4 構築の規模

データ辞書の登録は、システム再構築で利用するファイルを中心に行っている。現在の登録状況は表1のとおりであり、今後随時登録内容を追加していく予定である。

表1 データ辞書への登録状況

システム数	11
ファイル数	391
データ項目数	11,404
統一項目数	9,724
コードドメイン数	597
ドメイン体系数	6,626明細 (1ドメインにつき平均11明細)

なお、データ辞書システムのプログラムは、21画面、9モジュールと小規模に抑え最小限の工数に絞りこんで開発を行った。

その結果、主要機能部分は約10日ほどで完成し、データ整理作業を重点的にすすめることができた。(図7)

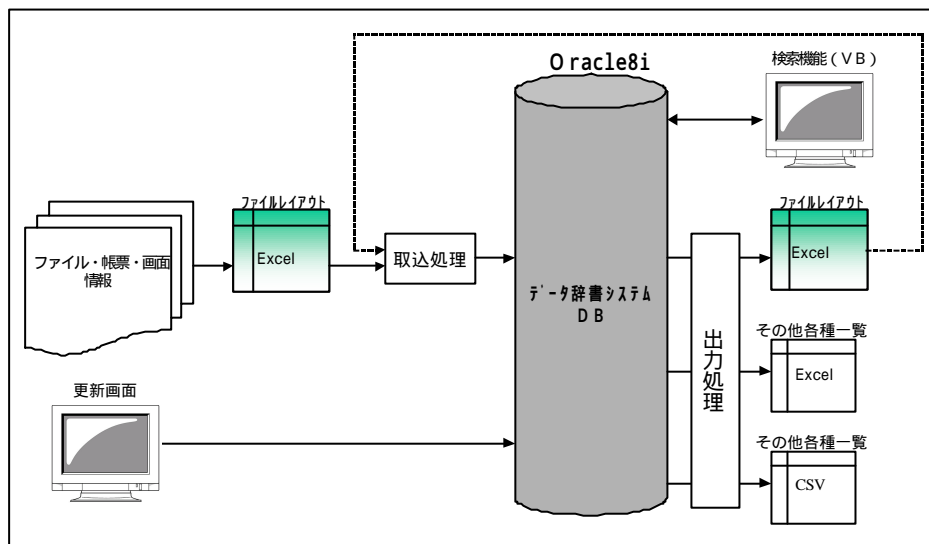


図7 アプリケーションの範囲

### 3.5 利用者側の処理

ログオン画面にユーザ名とパスワードを入力し、ユーザチェックが行われた後、メニュー画面が表示される。メニュー画面から、次の3機能をそれぞれ呼び出す仕組みになっている。

- (1) ファイルレイアウト出力機能
- (2) 検索機能
  - ・ 統一項目アプローチ
  - ・ コードドメインアプローチ
  - ・ データ項目アプローチ
- (3) その他管理機能(データのインプットやメンテナンス、ユーザー管理など)

本アプリケーションの最大の特徴は、検索機能でほかのアプローチと連携して高度な分析作業を可能としている点にある。以下にその具体例を述べる。

<例1：データ項目名「数量単位」と意味が重複する項目を調べる>

[検索]画面で全データ項目を対象に「データ項目名 = "数量単位"」という条件で検索を行う。このデータ項目「数量単位」につけられた統一項目名(まったく同じ内容・意味のデータ項目をグループ管理するための名称)が「数量単位区分」であることを確認する。(図8 - )

次に、同画面で「統一項目名 = "数量単位区分"」という条件で検索を行うと、同じ統一項目名がついているデータ項目が3件抽出される。(図8 - )

従って、同じ意味のデータ項目は「数量単位」自身を含めて3件あることがわかる。

**検索画面**

検索アプローチ：  
 データ項目   
 統一項目  
 コードメイ

システム名	ファイル名	データ項目名	統一項目名	属性	桁数	コードメイ名
XXX	XXX	数量単位	数量単位区分	X	6	数量単位区分(本社)

**検索画面**

検索アプローチ：  
 データ項目   
 統一項目  
 コードメイ

システム名	ファイル名	データ項目名	統一項目名	属性	桁数	コードメイ名
XXX	XXX	数量単位	数量単位区分	X	6	数量単位区分(本社)
XXX	XXX	数量単位区分	数量単位区分	X	4	数量単位区分(鹿島)
XXX	XXX	数量単位コード	数量単位区分	X	20	数量単位区分(ERP)

図8 データ辞書システムの利用例1

図8 - で表示されている統一項目情報を見ると、同じ統一項目名を持つデータ項目の桁数が必ずしも一致しないことがわかる。これは、事業所単位で作られた各システムで、例えばA事業所システムではデータ項目を余裕をもたせて6桁に設計し、B事業所システムでは4桁としたようなケースでも、同項目に対して管理可能にするための仕様である。

以上の仕様によって、(1)ファイルの正規化を行う際の強力な資料になる(2)正規化作業を行わなくても、正規化を行った結果を把握できるので、短時間でデータ分析を行うことができる。といったメリットがある。

<例2：「数量単位」で始まるデータ項目のコード体系を比較する>

[検索]画面で全データ項目を対象に「データ項目名 = "数量単位\*"」という条件で検索を行い、抽出結果をみるとコードドメイン名(コード体系が等しいデータ項目をグループ管理するための名称)も「数量単位区分」であることがわかる。(図9 - )

次に、同画面で「コードドメイン名 = "数量単位区分\*"」という条件で検索を行い、抽出されたコードドメインを一つずつ選択した状態で[コード体系]を表示する。(図9 - )  
 選択したコードドメインとリンクされたコード体系の詳細が一覧表示されるので、似たコードドメインのコードブック内容を比較することができる。(図9 - )

このように、コードドメインを通してコード体系の内容を調べることができるので、  
 (1) 新データ項目のコードドメイン名を新規に命名する際に、重複するコードドメインの増加を防ぐことができる (2) 新システムに移行する際にコードドメインを統合化する場合の資料にする といった利用方法が考えられる。

The image shows a three-step process in a software interface:

- Search Screen 1:** Search criteria: Data Item Name = "数量単位\*". Results table shows three rows with code domains: 数量単位区分(本社), 数量単位区分(鹿島), and 数量単位区分(ERP).
- Search Screen 2:** Search criteria: Code Domain Name = "数量単位区分\*". Results table is identical to the first screen.
- Code System Detail:** Selected code domain: 数量単位区分(本社). Attributes: 属性: X, 桁数: 6. A table lists unit codes and their content.

システム名	ファイル名	データ項目名	統一項目名	属性	桁数	コードドメイン名
XXX	XXX	数量単位	数量単位区分	X	6	数量単位区分(本社)
XXX	XXX	数量単位区分	数量単位区分	X	4	数量単位区分(鹿島)
XXX	XXX	数量単位コード	数量単位区分	X	20	数量単位区分(ERP)

区分	内容1	内容2
000001	ミリグラム	M
000002	グラム	G
000003	キログラム	KG
000004	トン	T

図9 データ辞書システムの利用例2

## 4. 主な工夫点

データ辞書システムを構築する際に、以下の3つの工夫を行った。

### 4.1 規約による標準化

ホストシステムでは半角英数字や記号の使用には制約がある。また、データベースを管理しているOracle 8iでは、内部的に特殊な扱いをしている文字があり、その文字をデータベース内に登録してしまうとVisual Basicアプリケーション実行時にエラーが発生してしまう。このような問題を回避するため、データベースへ登録する内容の規約書を作成し、ユーザに対し一定の文字を使用しないよう統一をはかり登録を行うことによって解決した。

また、規約を設ける以前は漢字の送り仮名の不統一（例：「積み合わせ」と「積み合せ」など）やカナの長音の表記不統一のために、データ辞書システムを検索してもヒット率が低かったが、この規約で表記の統一をはかったことによってヒット率を向上させることができた。

### 4.2 検索機能の充実

各項目名の検索などを行う際に正式名称を覚えているユーザは少なく「数量単位」や「ユーザ区分」というように、あいまいな名称しか覚えていないケースが多く、何度も表現を変えて検索をしないとヒットしないという問題があった。

そこで、検索条件を拡張し、検索文字との「完全一致検索」のほか、検索文字が含まれている項目をすべて抽出する「任意一致検索」や、検索文字で始まる項目をすべて抽出する「前方一致検索」、検索文字で終わる項目をすべて抽出する「後方一致検索」の4パターンを設けた。これによって、検索時のヒット率が格段に向上し、ユーザはあいまいな記憶を頼りに欲しい情報を抽出できるようになった。

### 4.3 ヒアリング内容のデータベース化

当社では、ユーザと開発者との間で業務用語に関する認識の違いが生じ、それによって手戻り作業がしばしば発生して問題となっていた。

そこで、エンドユーザからヒアリングした業務用語を開発者が登録するように運用ルールを取り決めて作業の標準化をはかったところ、意思疎通がスムーズになり、手戻り作業を低減させることができた。

## 5. 評価

データ辞書システムを社内で利用してもらい、エンドユーザとシステム開発者からの意見を以下にまとめた。

### 5.1 エンドユーザの評価

- (1) システムを再構築しデータ項目の名称が変更されても、現行の「業務名称」はデータ辞書システム中に残るので、辞書を参照すれば新しいデータ項目名がわかる。このことにより、新システムに移行されてもユーザ側には殆ど負担がかからないという安心

感・信頼感が生まれた。

- (2) エンドユーザがデータ項目名と意味を意識するようになり、今後のEUC活用へのステップになった。

## 5.2 システム開発者の評価

- (1) 開発の際の情報収集が省力化され、開発工数を低減させることができた。
  - 探しているデータ項目が、どのシステム・ファイルに存在するのかを簡単に調べられるようになった。
  - エンドユーザからヒアリングした業務用語を開発者が登録することによって、情報を開発者全体で共有できるのでエンドユーザ・開発者間の意志疎通不全からくる開発手戻りが減少した。
  - ファイルレイアウトがペーパーレス化し、データとして一括検索できるようになったことで、作業効率が向上した。
  - 最新のファイルレイアウトやコードブックがオンライン化され、調べやすくなった。
- (2) ファイルレイアウト、データ項目情報がそれぞれ一元管理されているので無駄なファイルやデータ項目を作成せずに済み、将来的に保守の負担軽減になる。
- (3) ファイルレイアウトやコードブックがペーパーレス化したことで、ドキュメント管理の負担が軽減し、経費が削減された。
- (4) 「命名規約書」によって、ファイル設計の社内標準化をすすめる体制が整った。
- (5) 命名規約に基づく各項目命名と多機能検索機能の効果によって、検索時にヒット率が高いので使いやすい。
- (6) データを正規化した結果に近い情報を取りだせるようになったため、過去に作成したファイルを正規化する前段階の資料となる。

## 6. 今後の展望

### 6.1 分社化及び他企業間インターフェース増加への対応

当社が保守・メンテナンスを行う親会社では、カンパニー制の採用及び機能別分社化を進めているため、サブシステム間同士をつなぐインターフェースは増加していくと予想される。また、グローバル化が進んでいる今、XMLなどが代表する、他企業間インターフェースへの対応も視野に入れる必要がある。

このような事態に備え、データ辞書適用範囲の更なる拡張と、他システム間のデータ項目分析機能をより強化させていく方針である。

### 6.2 Web化とXML展開

今回開発したOracle 8i - Visual Basicアプリケーションによるデータ辞書システムでは、ユーザに公開するには各クライアント端末にOracle接続ツールや開発したVisual Basicアプリケーションをインストールする必要がある。今後は分社化も視野に入れ、導入が簡単で負担の少ないOracle 8i - Webアプリケーションによるデータ辞書システム開発を検討していきたい。

また、このデータ辞書を利用したXMLシステム開発が実現すれば、EDI推進の強力な支援ツールとなる。この場合、「統一項目」の概念はXMLの「タグ」として、「コードドメ

イン」は属性・桁数・意味などが管理されているためデータチェックなどへの流用が考えられる。また、コードドメインはコード体系とリンクされているので、ユーザーが入力した値がその区分のコード体系に含まれる正しい値なのかどうかのチェックに利用することもできる。

## 7. むすび

今回のデータ辞書システム開発は、全社システム再構築からはじまったものである。

リスクの高い全プログラム自社開発、全ファイル正規化作業を回避し、短期間でグローバル社会に対応し得るデータ中心型の新システムを構築するという目標のために開発した本データ辞書システムは、今その成果を順調にあげはじめている。

我々は本システム開発の成功に満足するに留まらず、これを利用してシステム再構築という一大事業を成功に導くべく、より一層の努力をしていく所存である。

最後に、今回のシステム開発では富士通(株)殿の「CNAP - 」や「SDEM90」を参考にさせて頂き、最適化された開発工程で作業をすすめることができた。そして今回のシステム開発に対してこのような執筆の機会を与えてくださったFUJITSUファミリー会殿に、深く感謝申し上げます。