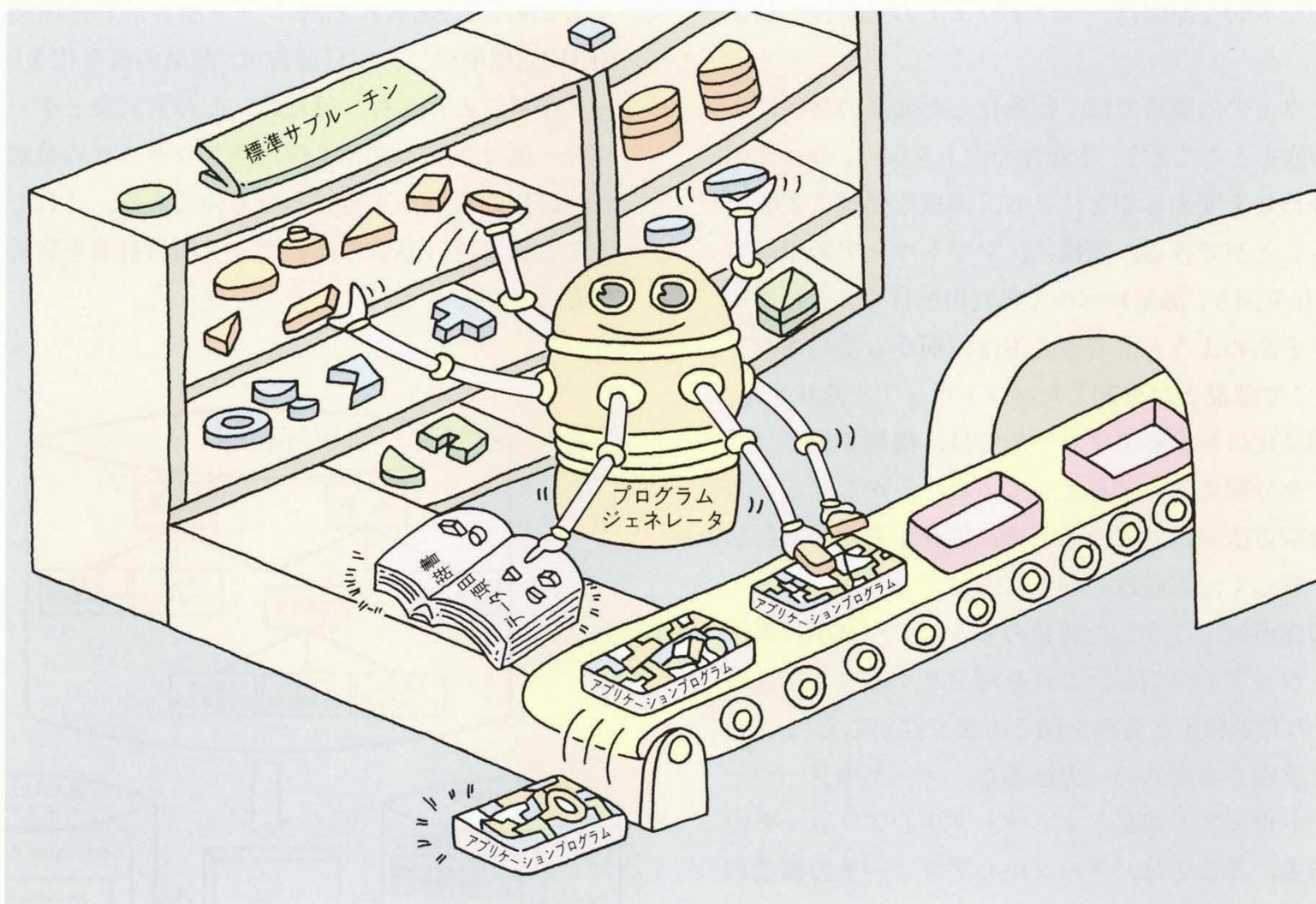


データ中心アプローチを応用した標準データ項目辞書の開発とその活用方法

Development and Application of Standard Dictionary Utilizing Data Oriented Approach

森岡 洋介* Yousuke Morioka

降旗由香理* Yukari Furuhashi



データ項目辞書を用いたアプリケーションプログラムの開発 データ項目辞書を用いると、標準化したソフトウェア部品を合成することによって、アプリケーションプログラムを容易に作成することができる。

ソフトウェア開発需要や保守コストの増大など情報システムが直面している問題の解決に対し、近年データ中心アプローチやオブジェクト指向と呼ばれる方法論がクローズアップされている。これは、データを情報システムでの部品と位置づけ、これに基づいてシステムおよびソフトウェアの設計を行う概念である。しかし、これを行うための手順、支援ツールが十分整備されておらず、実現されている例はまだ少ない。

日立製作所は、データ中心アプローチによるシステム開発をサポートするため、SEWB3(Software

Engineering Workbench 3)を開発した。その一連の製品のひとつであるSEWB3標準データ項目辞書は、汎(はん)用的なデータ項目を抽出し、これについて統一性を持った属性と処理を結合し、データのカプセル化を実現したものである。

このSEWB3標準データ項目辞書の活用は、業務システム用データ項目辞書の作成での生産性と品質の向上、さらに、これをベースとしたプログラム開発での生産性と品質の向上、および保守工数の削減に寄与すると確信している。

* 日立製作所 公共情報事業部

1 はじめに

業務システムの大規模化とこれに伴う保守コストの急速な増加、また、情報システムに対する要望の多様化と高度化に伴うバックログの増大など、現在の情報システムが直面している問題は多岐にわたっている。これらの諸問題の根本的な原因は、ソフトウェアの生産性の低さにある。

ハードウェアの製造では、標準化した部品の合成による生産形態をとることと、生産性の向上を図り、かつ部品の組み合わせを変えることにより、多様な要求に柔軟に対応することができる。同様に、ソフトウェアの生産でも、部品化を図り、部品レベルで再利用を行うことによって生産性を高めようとする考え方は以前から提唱されてきた。ここで部品とは分割したソフトウェアを意味する。従来、部品化に対するアプローチでは、機能単位でソフトウェアを分割する方法をとっていた。しかし、この方法では分割方法が設計者の考え方に依存するため、部品の共有が難しく、生産性の向上に寄与できなかった。

日立製作所は、こうした背景のもとにデータ項目単位でソフトウェアを分割し、これを部品とすることによって、部品の有効利用と管理を図る方法を採用した¹⁾。この考え方を実現するための一方法論が、データ項目一つ一つをソフトウェアの部品とするデータ中心アプローチの概念である。ここでは、データ中心アプローチの概念の実現を目指して開発したSEWB3(Software Engineering Workbench 3)標準データ項目辞書の概要、およびこれを用いたデータ項目辞書の作成方法について述べる。

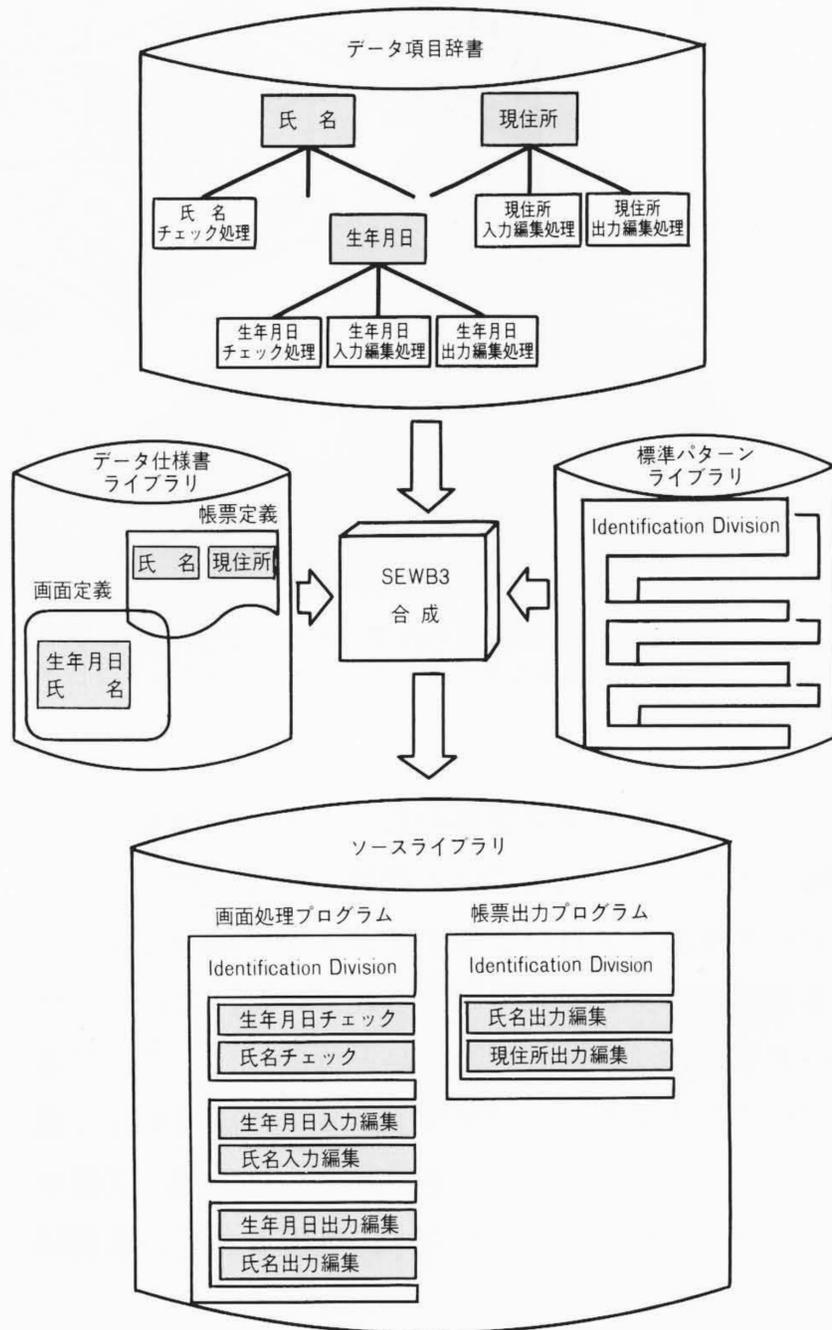
2 データ中心アプローチの考え方とその適用効果

2.1 データの標準部品化

これまで、部品はプログラム単位にその処理プロセスに基づいた機能分割によって設計されていた。機能の分割方法は設計者に依存し、「機能」はソフトウェアの開発工程、および開発担当者に応じて変わるため、ソフトウェア作成での部品の再利用が困難であった。これに対して、データ中心アプローチの考え方は、実世界の実体に従って存在するため、安定して不変である「データ」をソフトウェアの部品とする²⁾。そのため、システム構築での部品のレベルを統一でき、さらに部品設計をシステム設計の初期段階で行うことができる。この結果、共通した認識を持った部品化を実現できるため、部品の再利用が可能になり、ソフトウェアの生産性向上に寄与すると

期待できる。また、部品を組み合わせることにより、多様な機能を持つソフトウェアの作成に容易に対応することも可能となる³⁾。

日立製作所は、データ項目単位に、ソフトウェアの部品を作成し、これを合成してシステムを構築する方法を採用した。この方法によるソフトウェア開発環境を提供するために、SEWB3標準データ項目辞書を開発した。SEWB3標準データ項目辞書は、部品の標準化を図り、データ項目ごとに名称、けた数、記録形式などデータの属性と、個々のプログラムや処理プロセスから分離した汎(はん)用的な処理を定義したものである。これを活用することにより、品質の高いデータ項目辞書を容易に作成することができる。



注：略語説明 SEWB3 (Software Engineering Workbench 3)

図1 プログラム生成方式 画面や帳票の定義情報によって決定するデータについて、データ項目辞書からデータ部品を取得し、これを標準パターンに埋め込むことによってプログラムを生成する。

2.2 プログラム合成機能の改良

データを資源と見なして部品化を図るということは、単にデータだけを管理するという問題ではない。データを共有化して一元管理するならば、それらのデータを利用したプログラムの生成方法にも、それに対応した改良が必要である。

そのため日立製作所は、従来のプログラム合成機能を改良して、データ項目辞書を取り込む新しいプログラム合成機能をSEWB3で開発した⁴⁾。そのプログラム生成方法は、データ仕様書から入出力ファイルあるいは帳票に関する仕様情報を参照することにより、当該プログラムで必要なデータ項目を認識し、そのデータ部品の中の必要な処理を、あらかじめ決まっているプログラムパターンの所定の位置へ埋め込むものである。プログラム生成方法のモデル化を図1に示す。プログラム生成の手順は次のようになる。

- (1) 生成ターゲットのプログラムが扱う帳票や画面の定義情報から、生成するプログラムで必要なデータ名を決定する。

帳票定義…「氏名」「現住所」

画面定義…「氏名」「生年月日」

- (2) プログラムパターンライブラリから該当する標準パターンを選択し、取り込む必要があるデータ部品とプログラムパターン上の取り込む位置を決定する。

帳票プログラムデータ名…「氏名」「現住所」の出力編集処理

画面処理プログラム…データ名「氏名」「生年月日」のチェック、入出力編集の各処理

2.3 適用効果⁵⁾

データ項目辞書を適用した、実プロジェクトでのプロ

表1 プログラム合成機能の改良によるプログラム自動生成率とバグ密度の比較 プログラム合成機能の改良後では、作成したプログラムのバグ密度が改良前と比較して減少している。これは工程が進むほど顕著になる。

No.	項目	改良前	改良後	
1	生成率 (%)	68	81	
2	1プログラムの規模(ks)	0.65	0.58	
3	バグ密度	単体テスト(UT)	100	87
		結合テスト(CT)	100	73
		総合テスト(ST)	100	10
	合計	100	66	

注：略語説明 ks(キロステップ) UT(Unit Test) CT(Combination Test) ST(System Test)
バグ密度のデータ数値は、改良前のプログラム合成を用いて開発した場合のデータを100とした相対数値である。

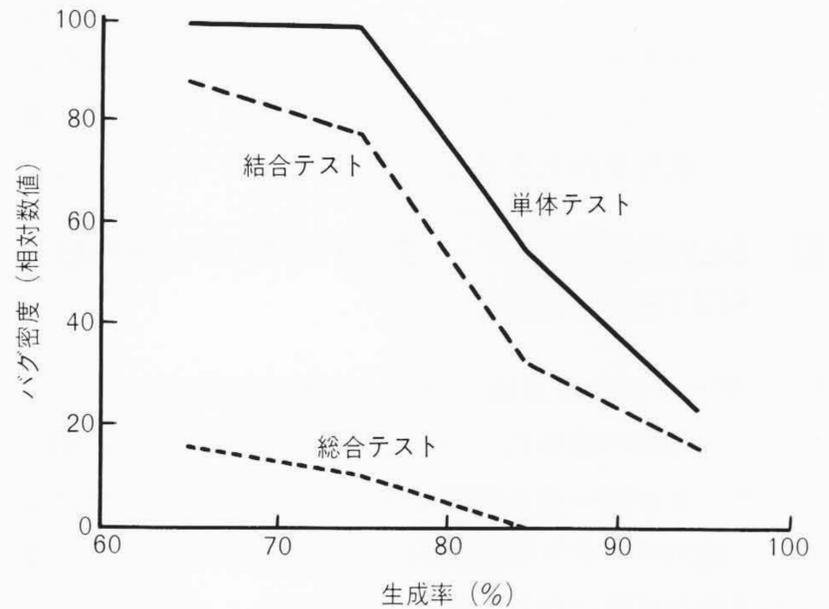


図2 プログラム生成率とバグ密度の関係 工程に関係なく、プログラム生成率の向上に伴って、バグ密度は減少する。特にプログラム生成率が80%を超えると、バグ密度の減少は著しい。

グラム開発の信頼性と生産性について評価する。

信頼性の評価では、プログラム自動生成率(以下、「生成率」と呼ぶ。)とバグ密度を調べた。改良前と改良後のプログラム生成ツールを用いて開発した生成率と、バグ密度を表1に示す。なお、バグ密度のデータの数値は、改良前のツールを用いて開発した場合のデータを100とした相対数値とする。

改良前のツールを用いた場合に比べて、改良後のツールを用いた場合の生成率の向上とバグ密度の減少を表1に示した。この結果は、生成率の向上に伴うプログラマによる修正ロジックの減少によるものと推測できる。さらに、改良前のツールを用いた場合のバグ密度との差は、単体テスト、結合テスト、総合テストと工程が進むほど大きくなっていることから、信頼性の向上は後工程に引き継がれ、むしろ工程が進むほど適用効果が顕著になると予想される。

次に、生成率とバグ密度の関係について調べた。生成率の変化に伴うバグ密度の変化を図2に示す。縦軸は生成率60%台のUT(Unit Test)時を100とした相対数値で表したバグ密度、横軸は生成率である。生成率の向上に伴って、バグ密度が減少している。最小二乗法による回帰分析を行ったところ、全工程で0.9以上の相関係数が得られ、バグ密度と生成率には強い負の相関があることがわかった。これにより、プログラム開発でのデータ項目辞書の適用は、開発プログラムの生成率を向上させることにより、信頼性を向上できると考えられる。

生産性の評価では、作業グループごとにプログラムの作成工数を調べた。改良前のプログラムジェネレータを

用いた場合に比べ、改良後のそれを適用した場合では、プログラム開発の生産性が1.3倍から1.9倍に向上した。これによって、プログラム開発でのデータ項目辞書の適用が、生産性を向上させることができると考えている。

3 SEWB3標準データ項目辞書(SEWB3/SDITEM)の開発

3.1 データの命名方法—データ名称の標準化—

データ名称の標準化とは、データを共有する人の中で同一データが同一名と認識できることである。これによって、他のデータとの識別および名称によるデータの意味の認識が可能になり、データを部品として共有することができるようになる。そのためには、データ項目の命名方法に系統的な規則を定め、名称が一貫性を持つ必要がある。SEWB3標準データ項目辞書では、データ項目名の語構成を「実体名—アトリビュート名—ドメイン名」のような規則として決めることにより、データ項目名の標準化を図った。

実体とは識別可能なもの、アトリビュートとは実体を持つ特性、つまり実体がどのようなものであるかを示す識別子であり、実体とデータを結び付ける修飾語である。ドメインは意味、範囲などデータの値域を示す語である。この語構成は、データは実体を持つアトリビュートに特定なドメインを対応づけたものという考えに基づいている。命名例を表2に示す。

表2 名称基準に従ったデータ項目命名例 データ項目名称は、「実体名—アトリビュート名—ドメイン名」の語構成となるように命名する。

No.	実体名	アトリビュート名	ドメイン名
1	討論会	開 催	年月日
2	討論会	座 長	氏 名
3	発表者	—	氏 名
4	発表者	所属部課	コード
5	発表者	入 社	年月日

3.2 データ項目名の選定

データ項目名の選定は、代表的な約200の業務アプリケーションの開発で、実際に設計されている約2万のデータ項目の分析に基づいて行った。その手順は次の三つの作業から成る。

(1) データ項目名の単語単位の分割と抽出

データ項目名を単語単位の分割し、出現頻度の高い単語を抽出する。単語とは、文および文節を構成する言語の最小単位である。

例 予定年月日→「予定」+「年月日」

(2) 名称基準にのっとった単語の分類

抽出した単語を、実体名、アトリビュート名、ドメイン名に分類する。分類は、データ項目名中の単語の出現位置(先頭、中間、語尾)により、基準を設けて行った。

アトリビュート名には、先頭および中間に出現する単語のうち、出現頻度が高いものを採用した。

ドメイン名には、末尾に出現する単語のうち、出現頻度が高いものを採用した。

例 「予 定」+「年月日」

「予 定」…アトリビュート名

「年月日」…ドメイン名

(3) データ項目名の作成

データの標準化は、名称の統一だけでなく、その標準的な操作内容を設計することである。そこで、データ項目名の作成にあたり、業務アプリケーションでのデータ項目名とその操作内容の関係を調べた。その結果、データの操作内容は「アトリビュート名+ドメイン名」によって、おおよそ決定できることが確認できた。そこで、データ項目名はアトリビュート名とドメイン名を組み合わせで作成した(表3参照)。

表3の例では、「電話年月日」は日本語として明らかに不適である。このようなデータ項目名は除いた。

この手順により、SEWB3標準データ項目辞書に登録するデータ項目として、1,600にも及ぶデータ項目を選定した。

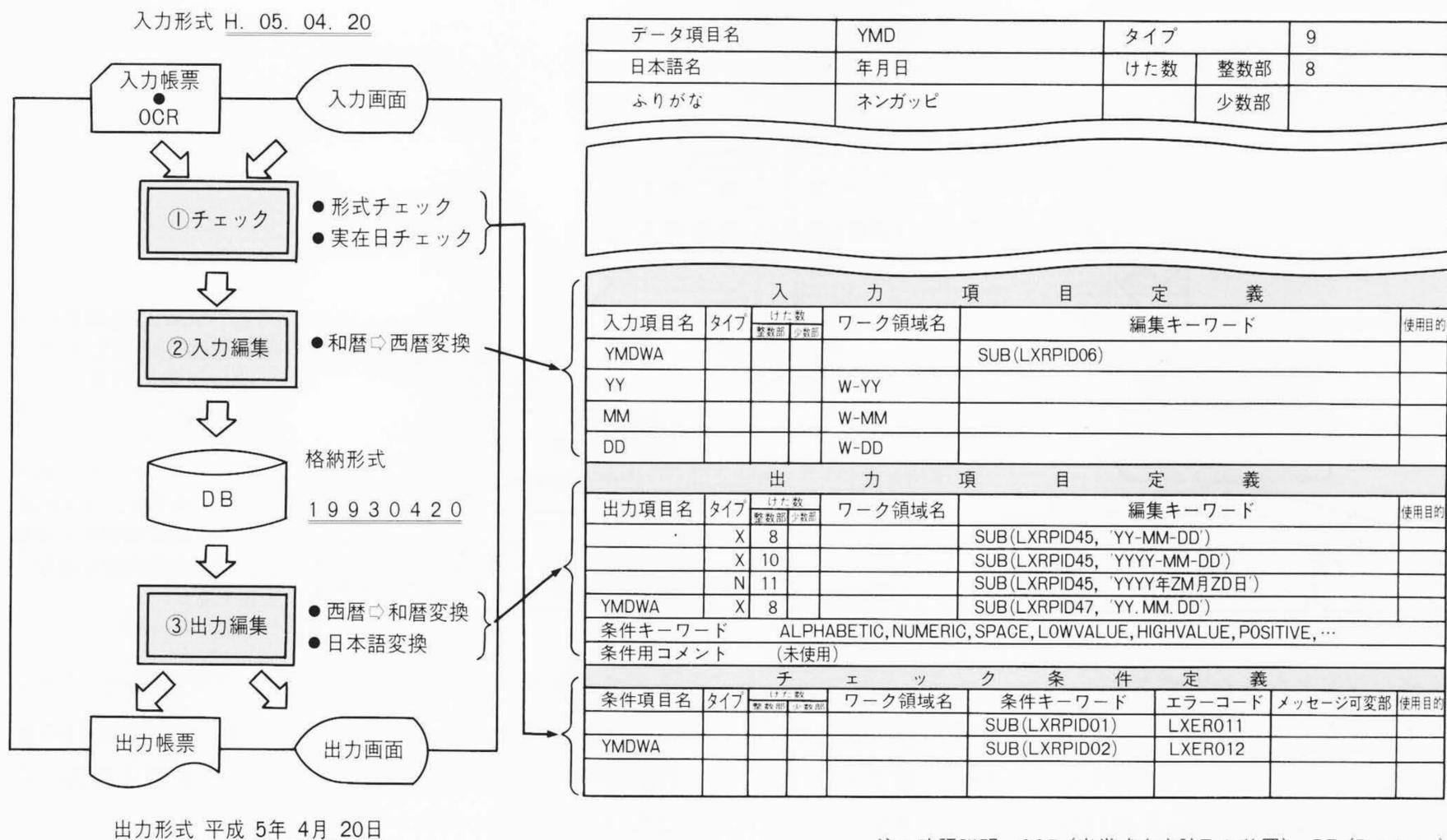
表3 データ項目名の作成 データ項目名は、アトリビュート名とドメイン名を組み合わせで作成する。しかし、日本語として不適なデータ項目名は除く。

	アトリビュート名	予 定	電 話
ドメイン名			
年 月 日		予定年月日	電話年月日
番 号		予定番号	電話番号

3.3 データと処理プロセスの一体化

選定したデータ項目には、アプリケーション、システム構成、業務内容などにかかわらず、業務アプリケーションでのデータの操作内容の分析に基づく経験則で処理を設定した。そのため、SEWB3標準データ項目辞書は、実際の業務アプリケーションでのデータの使われ方に準じたものになっている。しかし、業務アプリケーションでのデータの設定方法に問題があると判断した場合には、これに準じることなく対策を講じて内容を設定した。

データが受ける操作とSEWB3標準データ項目辞書の



注：略語説明 OCR (光学式文字読取り装置), DB (Database)

図3 SEWB3標準データ項目辞書の定義例 SEWB3標準データ項目辞書では、データ項目をDBの格納形式で登録し、データが受ける操作を、チェック、入力編集、出力編集の3処理に分類して定義する。

定義例を対応づけたのが図3である。同図の左側はシステム内で「年月日」というデータ項目に対して行われる処理をモデル化したものであり、右側はこのデータ項目のSEWB3標準データ項目辞書での定義例である。SEWB3標準データ項目辞書では、データ項目をDB (Database)の格納形式で登録し、データが受ける操作をチェック処理、入力編集処理および出力編集処理の三つに分類して定義している。

図3では、帳票あるいは画面を通して、システムに対し、入力形式 'H.05.04.20'の「年月日」のデータが入力されると、

- (1) その形式および妥当性をチェックする(チェック処理)。
 - (2) '19930420'という西暦日付の形式に変換する(入力編集)ために、西暦変換処理を行う
- 処理を行い、DBに格納する。出力する場合には、
- (3) 出力形式 '平成 5年 4月 20日'に変換する(出力編集)ために和暦および日本語変換処理を行う
- 処理を行い、帳票あるいは画面に出力する。

SEWB3標準データ項目辞書には次の二つの特徴がある。第一は、一つのデータ項目について複数処理の定義が可能である点である。これは、データの入力や表示が人間

の見やすさ、扱いやすさを考慮するものであり、データの入出力での編集方法は必ずしも1種類に限定できないという考え方に立脚したものである。第二は、設定する処理の大半がサブルーチンであるという点である。このサブルーチンは、SEWB3標準データ項目辞書に外部サブルーチンとして用意している。したがって、サブルーチンの設計・製造の工数の削減、およびプログラム開発の生産性と品質の向上を図ることができる。

4 SEWB3標準データ項目辞書(SEWB3/SDITEM)を用いたデータ項目辞書の作成方法

データ項目辞書の作成手順は、次の四つの工程から成る。

- (1) データの正規化
- (2) データ項目名称によるSEWB3標準データ項目辞書の検索
- (3) 検索したデータ項目辞書の定義内容の複写
- (4) 複写した定義内容の編集

データの正規化は、データ項目辞書の作成にとって重要である。それは、正規化により、データが単一の実体または実体の関連を表現したものになるため、データの重複および結合が最も少なく、データの独立性も高く、

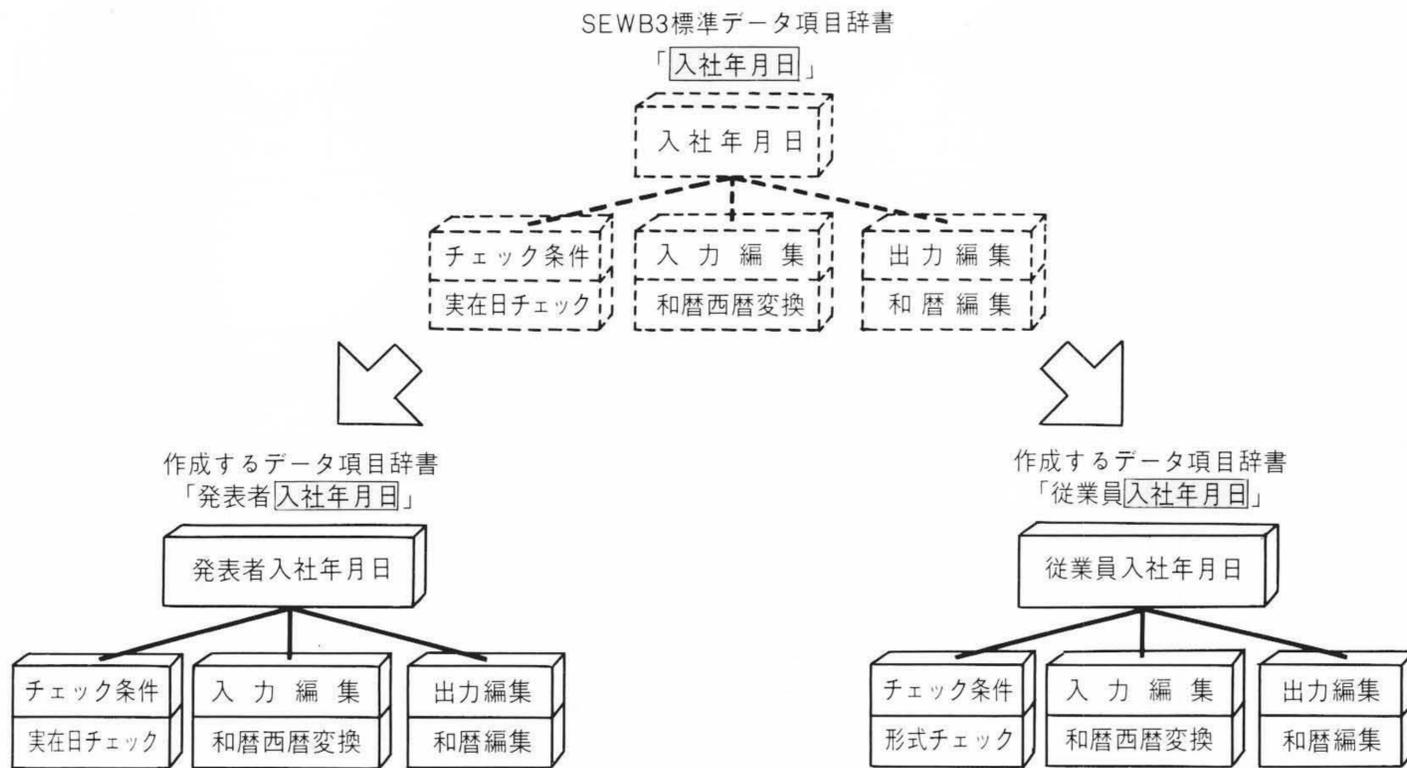


図4 SEWB3標準データ項目辞書を活用したデータ項目辞書の作成
生成ターゲットのデータ項目の「アトリビュート名+ドメイン名」が一致するデータ項目を、SEWB3標準データ項目辞書から検索し、その定義内容を複写、編集する。

適格な部品となるからである。

上記(2)から(4)の工程をモデル化したのが図4である。「発表者入社年月日」、「従業員入社年月日」の両データ項目は、SEWB3標準データ項目辞書からデータ項目名の「アトリビュート名+ドメイン名」が一致する「入社年月日」を検索できる。そこで、「入社年月日」データ項目の定義内容を複写し、それぞれ「発表者入社年月日」、「従業員入社年月日」のデータ項目の辞書として作成する。ここで、「従業員入社年月日」データ項目のチェック処理がSEWB3標準データ項目辞書から複写した「入社年月日」データ項目の仕様と異なる場合には、使用者が修正を行ってデータ項目辞書を完成させる。このように、SEWB3標準データ項目辞書を用いることにより、データ項目辞書を容易に作成することができる。

SEWB3標準データ項目辞書をより効果的に活用するには、データ名称の「アトリビュート名+ドメイン名」が一致するものを検索することである。SEWB3標準データ項目辞書では、「アトリビュート名+ドメイン名」に

よってデータの操作内容を設定した。したがって、「アトリビュート名+ドメイン名」の一致は、データ項目の操作内容の一致を意味し、目的とするデータ項目辞書に最適なデータ項目の検索ができるのである。

5 おわりに

ここでは、日立製作所が開発したデータ項目辞書の作成を支援するSEWB3標準データ項目辞書の開発思想と、これを用いたデータ項目辞書の作成方法について述べた。

データ中心アプローチの考えを適用するには、その基盤となるデータを管理するデータ項目辞書が不可欠である。SEWB3で取り組んだデータ中心アプローチは、データ項目レベルの設計での部品的な適用であり、それが信頼性、生産性に大きく寄与できることがわかった。今後は、データ中心アプローチさらにはオブジェクト指向に基づくシステム開発をよりいっそう推進する中で、データ項目辞書作成支援の強化を図っていく方針である。

参考文献

- 1) Y. Morioka, et al. : "Productivity Analysis of Software Development with an Integrated CASE Tool" 14th International Conference on Software Engineering pp.49~58(1992)
- 2) 堀内：データ中心システム設計，オーム社(1988年3月)
- 3) David A. Taylor, Ph. D. : "Object-Oriented Technology : A Manager's Guide" Addison-Wesley Publishing Company, Inc. (1992)
- 4) 森岡，外：EAGLE/Pデータ中心アプローチ支援ディクショナリの開発，情報処理学会第41回全国大会，4G-8，5/205~5-206(1990)
- 5) 降旗，外：EAGLE/Pディクショナリを用いたプログラムジェネレータの開発と適用効果，情報処理学会第43回全国大会，1K-1，5-381~5-382(1991)