

リレーショナルデータベースを用いた品質情報システム

Quality Information System through
Relational Data Base高 木 茂⁽¹⁾ 浦 本 明 博⁽²⁾
Shigeru Takagi Akihiro Uramoto

要 旨

リレーショナルデータベースを用い、品質情報システムを構築した。当社では品質関連情報を、部品受入検査、製造工程、出荷検査、市場に至るまで多くのステップで採取している。従来、これら品質関連情報は担当部署に分散して保管、解析がなされてきた。今回のシステム構築により、これら分散していた品質情報をホストコンピュータ上に一元化することができた。また、パソコン・ホストコンピュータ間のオンライン接続、リレーショナルデータベースの採用により、情報および情報の操作を解放したため、エンドユーザは品質情報に自由に、かつ容易にアクセス可能となった。システム開発に当っては、メーカー提供のソフトウェアコンポーネントの組み合わせでほとんどの処理を実現させたため、短期間でシステム開発ができた。

本稿では、先に品質情報システムにつき論じ、次に当社品質情報システムにつき、その概要を紹介する。

The relational data base system has been adopted for our quality information management system. The system has a network of the central host computer and a dozen of personal computers located in individual offices.

For the use of statistical analysis of quality information, our quality information management system accumulates various source data of a couple of years on many stages, such as parts inspection, manufacturing process, outgoing inspection, field data, and so on, and engineers in individual offices can get their necessary information easily, which are sorted out by relational treatment of files.

The article discusses the features of relational data base system and the application for our quality information management system.

(1), (2) 品質保証部

1. ま え が き

現経済環境下、競合他社との熾烈な競争、年々高まる顧客の要求品質水準、NIESからの廉価製品による追い上げ等、当社を取り巻く環境は内外ともに厳しい状況にある。当社ではこれらに対応するため、業界で品質面においても常に優位性を保つべく、従来から品質向上のため種々の活動を実施してきた。これら活動の推進には品質情報を的確に掌握管理の上、分析することが必要不可欠である。また、経営情報システムとしての品質情報の重要性も年々増加している。

当社では品質情報を、部品受入検査工程、組立製造工程、出荷検査工程、市場に至るまで多くのステップで採取している。従来、それぞれ担当部署に分散して保管、解析がなされていたこれら情報を、今回品質情報システムを構築することにより、ホストコンピュータ上に一元化することができた。

本稿では先に品質情報のあり方を論じ、更に当社が構築した品質情報システムの概要を紹介する。本システムは、第1次計画分が完了し、現在第2次計画分を推進中であり、未だ発展途上にあるが、本稿では第1次計画分につき紹介する。

2. 従来の品質情報

2.1 従来の品質情報の問題点

当社でいう品質情報とは、最終製品および最終製品を構成する部品の 1)不具合の状態を表わす情報または 2)良品の度合を表わす情報のことであるが、狭義には前者を示す場合が多い。本稿では、この両者を合わせ品質情報という。これらの目的は、異常品質の検知、不具合原因の改善効果確認等である。

一般に情報の良し悪しを問う尺度としては、次の5項目が考えられる。

- 1) 情報の精度（正確さ）
- 2) 情報の量
- 3) 情報の伝達速度（情報発生時と使用可能時のタイムラグ）
- 4) 情報の表現形式
- 5) 情報の集中度

システム構築に当り、事前にアンケート調査等も交え綿密な事前調査を実施したが、当社の場合も品質情報の問題点は、上記5項目に集約される。個々の詳細な問題点については列挙しないが、特徴的なユーザサイドの要求（アンケート調査結果でも高率であった）を挙げると、次の通りである。

- 1) 情報を自分で、自分の欲しい時に、欲しい型で入手したい
- 2) 試行錯誤的解析を行いたい
- 3) 情報に容易にアクセスしたい

これらができないということが、品質情報の現状の問題点であり、情報の「表現形式」に問題点があることが判った。

市場品質情報の事例で補足説明する。市場品質情報については、担当部署が月報形式で詳細な、加工されたデータ（グラフ、クロス帳票、一覧表等）を提供しているが、更に次元の異なる解析が必要な場合には、事前に視点が固定されたペーパー情報であるが故に、情報の再加工または、連鎖的に情報を取り出すこと（試行錯誤的解析）が困難または、不可能となっている。対応策として、EDP化されている情報については、データ処理部門にプログラムの作成、帳票の出力を依頼するわけであるが、バックログの関係もあり、数日～1週間掛かるのが常であり、時代のニーズに合致していない。（欲しい時に即、情報を得るのが現代のニーズである。）ましてEDP化されていない情報については、原始帳票を手作業で処理するはめに落ち入るのが実情である。

2.2 リレーショナルデータベースの導入

既述の通り、ペーパー情報のように既に加工されてしまった情報では、今日の多様化する品質情報のニーズには応えられない。例えば項目(カラム)数が100個ある場合、ここから2個(例えば2次元のグラフの作成)の項目の組み合わせを選ぶと、 $100C_2=4950$ 通りもの組み合わせが存在し、それぞれに対応した加工情報を予め準備しておくことは、無論不可能である。このため、加工していない原始データをデータベース上に蓄積しておき、情報の利用時点で初めて、ニーズに合致した加工形態を決定し、情報を取り出すことが必要となってくる。

すなわち、品質情報の特徴としては、下記が考えられる。

- 1) 画一化した処理や蓄積した情報を単に検索するだけでなく、製品別、地域別等の観点で集約したり、試行錯誤的な情報検索が要求される。
- 2) 情報の集約加工の観点は、利用者のニーズによって、また時代の流れによって大きく変化する。

換言すれば、蓄積する情報の構造と、取り出す時の情報の構造が事前に明確に規定できないのが、品質情報の特徴である。

これらのニーズを満す手法を調査検討した結果、現時点においては、多少問題点はあるもののリレーショナルデータベース(以下RDBという)

が最適であるとの結論を得た。以上により、当社RDB利用の先駆として、品質情報システムにRDBを導入し試行錯誤を重ねてきたので、その経緯を報告する。

3. 情報と情報システム

3.1 情報の発生と情報システム

情報とは、データを表現するために用いた約束^(JIS)に基づいて、人間がデータに割り当てた意味のことである。品質情報システムのモデルを図-1に示す。当モデルは、製品または部品の不具合原因を、解析者が調査解析のうえ、コンピュータに入力し、第三者がその情報を利用する過程を示している。図から判るように解析者自身は、測定器が発生する情報の吸収源であると同時に、この情報に、自身が持っている知識、経験を加味し、不具合原因を特定し、これを新たな情報として発生させる発生源でもある。従って解析者の能力、センスにより発生する情報内容も異なるのである。またコンピュータも解析者に対しては、情報の吸収源として振る舞い、情報の利用者に対しては、情報の発生源として振る舞う。このように解析者、コンピュータとも情報を吸収し、かつ情報に解釈を加え新たな情報を発生させるので、それ自身が情報システムであり、これらの集まりもまた情報システムと考えられる。このように、情報は必ず情報システムから発生するものであり、情報シ

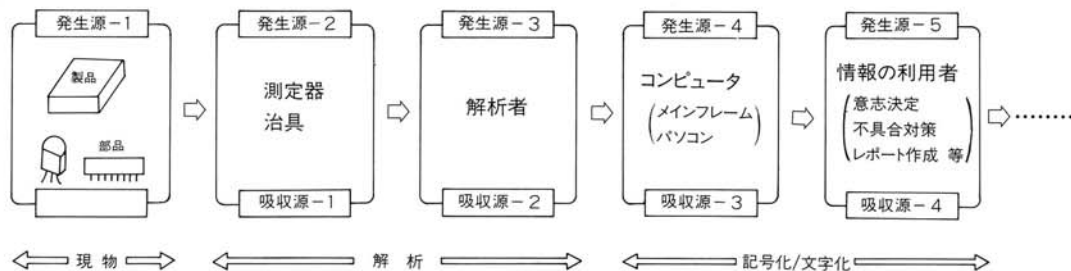


図-1 品質情報システムモデル

Fig. 1 Quality information model.

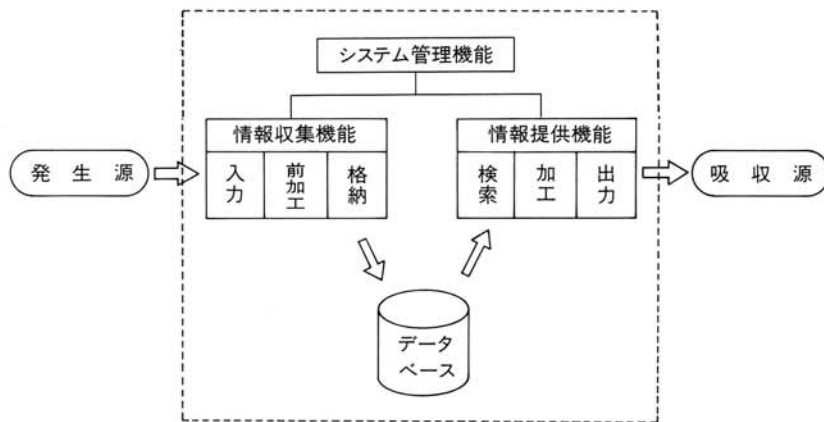


図-2 品質情報システムにおけるコンピュータの役割

Fig. 2 Assignment of a computer on quality information system.

システム無しには、情報は発生しえないのである。

なお、品質情報の内容を更に向上させるためには、これら測定器、解析者自身等の情報システムの精度向上を図ることも重要であり種々活動を行っているが、本稿では主に、狭い意味でのコンピュータを中心とした品質情報システムの範囲を扱っている。

3.2 品質情報におけるコンピュータの役割

図-1からも判るように、品質情報システムにおけるコンピュータの役割は、情報の発生源と吸収源の仲介をすることに他ならない。具体的には、図-2に示したように、情報収集機能、情報提供機能、システム管理機能である。また、情報をコンピュータに蓄積するということは、2次的に、情報の発生時点と利用時点におけるタイムラグを補

うという機能も持たせているのである。

3.3 情報システムの分類

情報システムの分類は、システムを特徴づける要素が多数あり、多くの分類方法が考えられるが、ここでは要求パターンの多様性と、処理加工の複雑性の2つの要素の組み合わせにより、図-3のように分類する。

1) タイプⅠ

要求パターンは予め決っており、処理加工も比較的単純なシステムである。一般事務処理等はほとんどこのタイプに属する。

2) タイプⅡ

要求パターンは予め決っているが、処理加工は複雑で多岐にわたるシステムである。科学技術システム等がこのタイプに属する。

3) タイプⅢ

要求パターンは多様であるが、処理加工は簡単なものや、出力編集程度のシステムである。文献情報検索システム等がこのタイプに属する。

4) タイプⅣ

要求パターン、処理加工とも事前に定まらない複雑なシステムである。経営意志決定システム等がこのタイプに属する。

品質情報システムの場合は、前記したように、

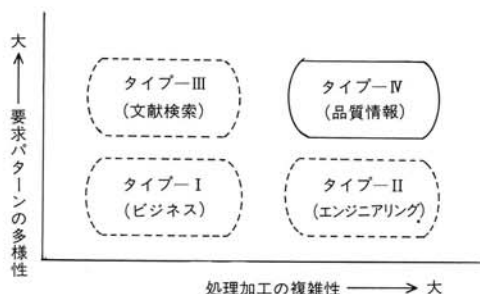


図-3 情報システムの分類

Fig. 3 Classification of applied fields.

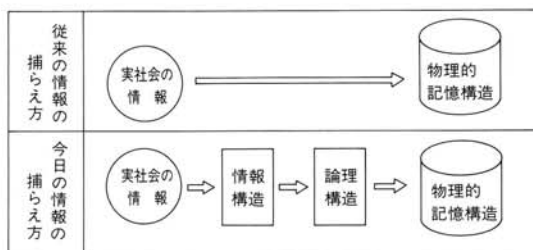


図-4 新・旧の情報の捕らえ方

Fig. 4 Conventional and modern logical recognition.

要求パターンは、利用者および時の流れにより様々に変化するため定まらず、処理加工の方法も試行錯誤的な検索が多いため、図-3の分類によるとタイプⅣに属する。

3. 4 情報の捕らえ方

情報をコンピュータに蓄積するためには、情報を論理的意味や物理的形式により規約化する必要がある、この規約化が情報の捕らえ方である。従来の情報の捕らえ方と今日の情報の捕らえ方は、¹⁾情報構造の扱い方において大きな相違がある。これを図-4に示す。

3. 4. 1 従来の情報の捕らえ方¹⁾

従来の情報の捕らえ方は、コンピュータのハード、ソフトの能力の問題もあり、コンピュータ特有の捕らえ方を、情報に強制していた。つまり実社会の情報が本来持っている情報構造を捕らえていなかった。これは実社会の実情報を、コンピュータの物理的記憶構造に当てはめなければならず、情報構造の構成要素である論理関係が捕らえられていなかったためである。

従来の目的従属型のファイルを今日の観点から見ると、次のような問題点がある。

1) 情報の冗長性

目的別にファイルが作成されるため、各ファイルに類似情報が重複して存在する可能性が大きく、資源的に無駄が生じる。

2) 要求パターンの多様化に対応不可

ファイルが目的毎に固定されているため、試行錯誤的な検索等の多様な要求には対応不可能である。

3) システムの変更困難

ファイルと処理が互に従属関係にあるため、一方に変更の必要が生じた時、他方も変更する必要がある、システムの変更、拡張が困難である。

3. 4. 2 今日の情報の捕らえ方¹⁾

コンピュータ技術の発展および、処理要求の多様化に伴い、情報の捕らえ方も変化してきた。従来の捕らえ方では配慮されなかった情報構造に着目し、その関連性から情報構造全体を捕らえようとするのが、今日の情報の捕らえ方である。

従来のファイルと根本的に異なる点は、情報が本来持つ情報構造の構成要素である論理関係までも蓄積する点にある。特定の目的に従属した専用ファイルではなく、情報が本来持つ論理関係をデータベースに持たせることにより、多様化する処理要求に対応できる、汎用性を持った情報の捕らえ方である。

4. データベース

4. 1 データベースの歴史的発展

今日のデータベースの思想に通じる商用システムが登場したのは1960年代初頭である。1970年代には構造型データベースが普及期を迎え、1968年 Dr. E.F.Coddが提案したRDBは現在、普及・技

脚注 1) 情報構造：情報が本来持つ相互関連をもとに、体系づけられた情報群

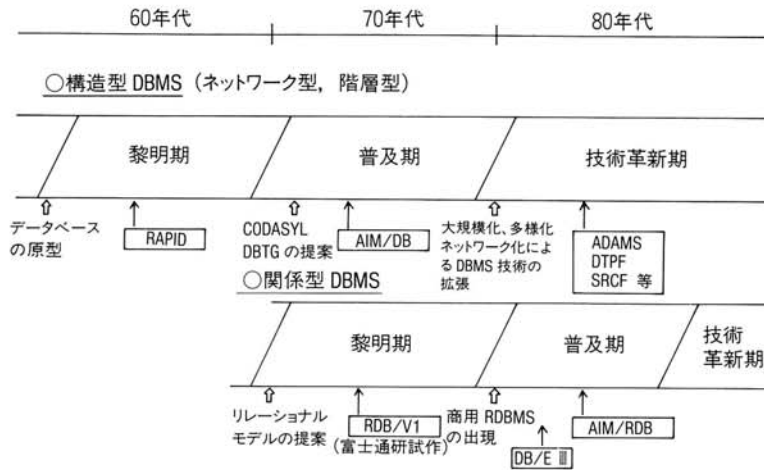


図-5 データベースの歴史と技術動向 (富士通の場合)

Fig. 5 Progress of data base system (FUJITSU).

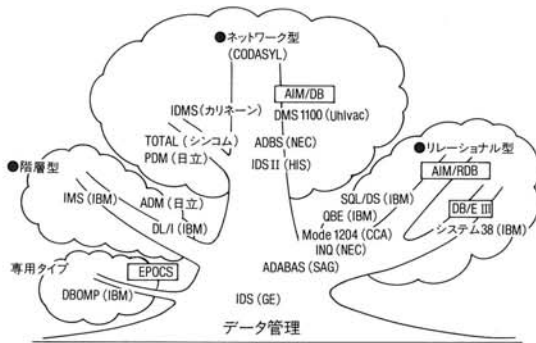


図-6 データベースシステムの系列

Fig. 6 Tree concept of data base families.

術革新期を迎えている。(図-5, 図-6)

4.2 データベース

データベースとは、不統一に管理されていた情報を、コンピュータにより一元的に管理し、多くのユーザが利用できるように統合化された共用ファイル群のことである。先に述べたように、従来のファイルが、利用目的毎に作成されたのに対し、データベースは、利用目的から独立して作成されるため、情報の一元化、応用プログラムの汎用化が図れることになる。これを図-7に示す。

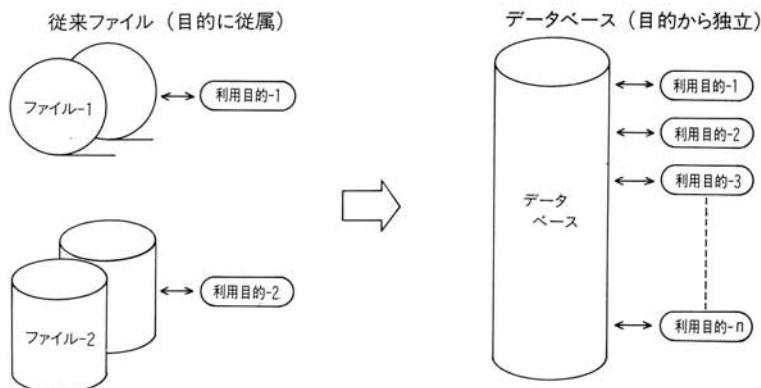


図-7 従来ファイルとデータベース

Fig. 7 Conventional file system and data base.

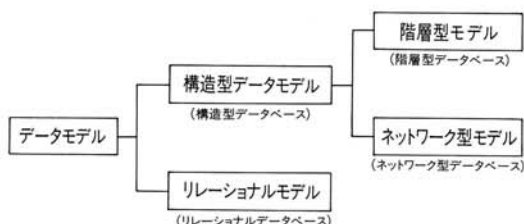


図-8 データモデルによる分類

Fig. 8 Classification of data model.

4. 3 データモデル

データベースに格納する情報は、実社会の情報をそのまま忠実に反映すべきであるが、現状のコンピュータでは、これは不可能であり、何らかの抽象化が必要となる。このように実社会の情報を抽象化した模型がデータモデルである。

今日ではデータモデルは、図-8に示すように、階層型モデル、ネットワーク型モデル、リレーショナル型モデルの3種に分類するのが一般的である。次にこれらモデルにつき、概要を述べる。

4. 3. 1 階層型モデル

実社会を構成する対象間の関係の大半が、階層構造で表現できるという考えに基いたデータモデルである。データ間の正順関係が1対 n 、逆順関係が1対1のTree構造で表わされる。これは、次に示すネットワーク型モデルで逆順関係に制限を加えた構造と考えられる。(図-9)

4. 3. 2 ネットワーク型モデル

図-10に示すように、階層型モデルで上位が複数個存在することを許したモデルである。データ間の正順関係が1対 n で、逆順関係も1対 m で表

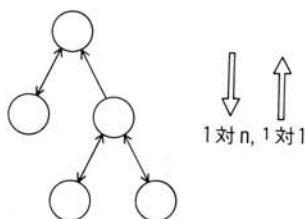


図-9 階層型モデル

Fig. 9 Hierarchical model.

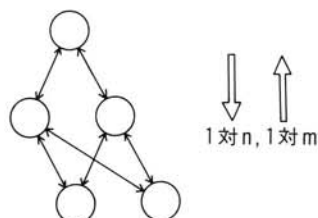


図-10 ネットワーク型モデル

Fig. 10 Network model.

わされる。

階層型モデル、ネットワーク型モデルを基本とする構造型データベースは、一般的に物理的なポインタにより情報間が結びつけられているため、RDBに比べ高性能が得られる。その反面、論理構造が利用者から見える形態のため、利用者はデータベースの構造を意識して、データベースを操作しなければならない。

4. 3. 3 リレーショナル型モデル

1968年、Dr. E.F.Coddが提案したモデルでRDBの基礎となっている。データは、図-11に示したようにテーブル(2次元の表)で表現する。前述した構造型モデル(階層型、ネットワーク型)と異なる点は、事前にデータ間の関係づけを行っていないという点である。データ間の関係づけはデータ操作時に随時行う論理構造をとったモデルであり、これは論理的表現と物理的表現が互に独立性を保っており、データベース利用者は、論理構造、物理構造を意識することなく、データベースの操作を行うことが可能であることを意味し

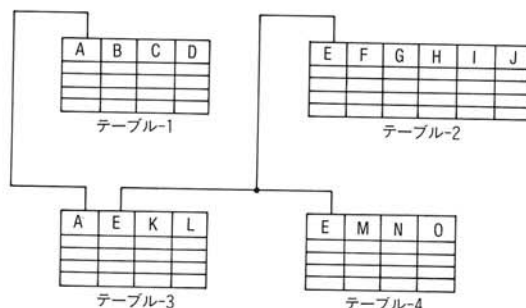


図-11 リレーショナル型モデル

Fig. 11 Relational model.

表-2 「真のリレーショナルデータベースが満すべき12の規則」⁽⁴⁾

ル		ー	ル	内	容
1	情 報 に 関 す る 原 則			データベースの中にあるすべての情報（データや表/行/列の名称など）を、ただ1つの形式（表形式）で表現する。	
2	ア ク セ ス の 保 障			表/行/列の名称とその値の組合せにより、すべての情報にアクセスできる（アクセスできない情報があるてはならない）	
3	空白値(ヌル値)の取扱			入力されていない情報（項目）を矛盾がおきかないような統一された方法で扱う（空白値を*たぶんXXX*として扱う）	
4	オンライン・カタログ			表/行/列の名称などデータベースを定義する情報を、データをアクセスするのと同じ言語（例えばSQL）でアクセスできる形式で表現	
5	包 括 的 な 言 語			データベースのもつすべての機能を実行できる言語（例えばSQL）が少なくとも1つはある	
6	ビ ュ ー の 更 新			いくつかの表を合成（join）して作った仮想表（ビュー）を正しく更新したら、その結果を合成する前の表に反映させる	
7	高 水 準 言 語			複数のデータ（レコード）を同時に挿入/更新/削除できる言語をもつ	
8	物 理 デ ー タ の 独 立			データの物理構造を変更してもアプリケーションに影響しない	
9	論 理 デ ー タ の 独 立			データの論理構造（表、ビュー）を変更した時のアプリケーションへの影響範囲を最小にする	
10	保 全 性 の 独 立			データのインテグリティ（首尾一貫性）を保つための条件を変更してもアプリケーションに影響しない	
11	分 散 か ら の 独 立			データを分散（あるいは集中）させてもアプリケーションに影響しない	
12	破 壊 さ れ な い 保 障			（データをレコード単位で扱うような）低水準言語でデータベースをアクセスしてもデータの首尾一貫性が失われない	

表-1 ネットワーク型とリレーショナル型DBMSの⁽³⁾比較

	ね	ら	い	特	徴
ネットワーク型DBMS	▷データの構造に沿った効率的アクセスを重視 ▷データの冗長性(重複)を極力少なくする			▷ある程度内部的構造（物理的な配置やデータの格納順序）を意識する必要があるが、ファイル効率、アクセス効率の良いデータベース構築が可能 ▷データ間の関係付けは、あらかじめ決めておく必要がある（効率的ではあるが柔軟性に欠ける）	
リレーショナル型DBMS	▷データベース構築を容易にする ▷データベース処理の単純化			▷データの物理的構造（配置、格納順序）を意識する必要はなく、単純でわかりやすい ▷データ間の関係付けは、データの値を基に行い、システムの拡張性、柔軟性に富む	

ている。RDBの特徴は次の通りである。

1) データの独立性が高い

データベースの論理的表現と物理的表現の分離

2) 構造が単純である

2次元の表で表わされ、数学的な集合論に基づく構造である

3) 高水準のデータ操作言語を備えている。

非手続型言語（AQL^{脚注2)}）を使用

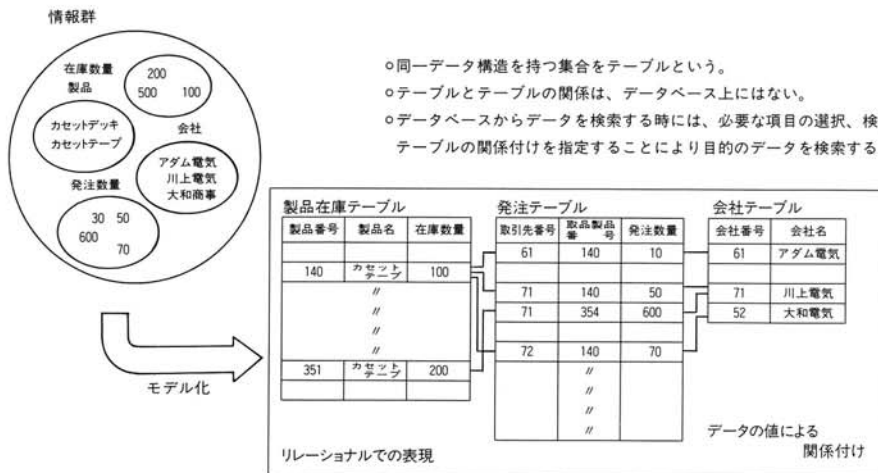
品質情報システムの今後の利用形態は、既述のように試行錯誤的解析が主流となるため、RDBのこのような柔軟なデータ構造が必要不可欠なものとなる。

ネットワーク型とリレーショナル型DBMSの^{脚注3)}

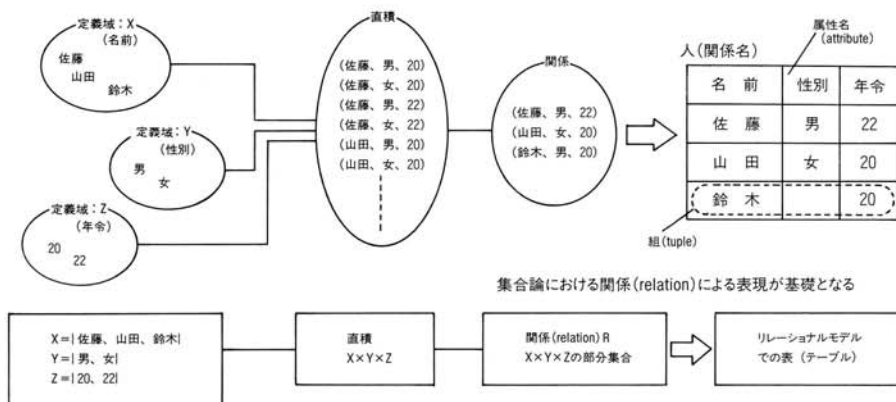
脚注 2) AQL: advanced query language (富士通 AIM/RDBの操作言語)

脚注 3) DBMS: data base management system

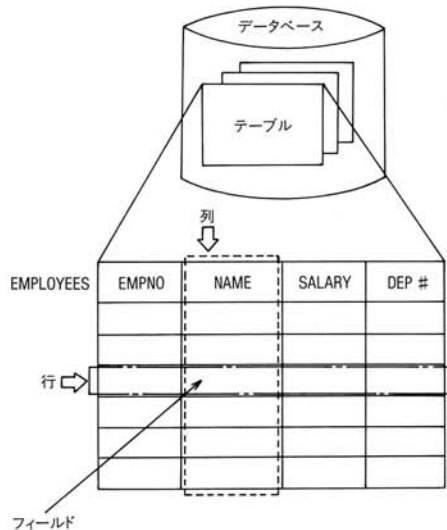
1. リレーショナルデータベースでの表現方法



2. リレーショナルモデルの考え方



3. テーブル形式でのデータ表現



リレーショナルデータベースは、テーブルという論理的ビューが基本であり、物理構成の意識は一切不要

高度なデータ独立性

テーブルとは、関係の集合論を具体的に表現したもので、

- ① テーブル、列には名前が付けられている。
- ② 一つのテーブルには同じ行がない。
- ③ 行及び列の出現順序は自由である。
- ④ 一つのフィールドには一つの値しか入らない。

などの特長がある。

ベース(RDB)とは

4. 必要な行だけのデータを取り出す(selection)の例

EMPLOYEES				
EMPNO	従業員番号	NAME	名前	DEP#
				所属番号
▷ 650064	佐藤太郎	250000	2000	
▷ 700128	久保 登	200000	1010	
▷ 750256	南 和男	175000	3110	
	770512	山本貞治	150000	2000
	810768	北島秀明	140000	1010
	820100	山口明子	132000	3110

SELECT EMPNO, NAME, SALARY, DEP #				
FROM EMPLOYEES				
WHERE SALARY >= 160000				
従業員番号	名前	給与	所属番号	
650064	佐藤太郎	250000	2000	
700128	久保 登	200000	1010	
750256	南 和男	175000	3110	

ある条件(給与160,000円以上)を満足する行を取り出す。

もとの関係(テーブル)のうちである条件を満足した組(行)だけを取り出し、新しい関係(テーブル)を作り出す操作を選択(Selection)と呼ぶ。

5. 必要な列だけのデータを取り出す(projection)の例

EMPLOYEES				
EMPNO	従業員番号	NAME	名前	DEP #
				所属番号
650064	佐藤太郎	250000	2000	
700128	久保 登	200000	1010	
750256	南 和男	175000	3110	
770512	山本貞治	150000	2000	
810768	北島秀明	140000	1010	
820100	川口明子	132000	3110	

SELECT NAME	
FROM EMPLOYEES	
(抜出し条件を付ける事も可)	
名前	
佐藤太郎	
久保 登	
南 和男	
山本貞治	
北島秀明	
川口明子	

必要な列(名前)だけを取り出す。

関係(テーブル)から不要な属性(列)を削除して、必要な属性(列)だけを取り出す操作を射影(Projection)と呼ぶ。

6. テーブルを結合してデータを取り出す(join)の例

EMPLOYEES				
EMPNO	従業員番号	NAME	名前	DEP #
				所属番号
650064	佐藤太郎	250000	2000	
700128	久保 登	200000	1010	
750256	南 和男	175000	3110	
770512	山本貞治	150000	2000	
810768	北島秀明	140000	1010	
820100	川口明子	132000	3110	

DEPT	
KNO	DNAME
課コード	部門名
1010	営業課
2000	企画課
3110	経理課
4355	総務課

SELECT EMPNO, NAME, DNAME			
FROM EMPLOYEES, DEPT			
WHERE DEP # = KNO			
従業員番号	名前	部門名	
650064	佐藤太郎	企画課	
700128	久保 登	営業課	
750256	南 和男	経理課	
770512	山本貞治	企画課	
810768	北島秀明	営業課	
820100	川口明子	経理課	

所属番号と課コードの値が一致する行を取り出し、一つのテーブルとする。

(等価結合)

複数のテーブルを共通となる列(所属番号と課コード)をもとに貼り合わせ一つのテーブルにまとめる。

二つの関係(テーブル)を共通となる定義域(列)をもとにして結合し、新しい関係(テーブル)を作り出す操作を結合(Join)と呼ぶ。

比較を表-1に示す。またDr. E.F.Coddが提案した³⁾「真のリレーショナルデータベースが満すべき12の規則」を参考のため表-2に示す。(P.59 参照)

5. 当社品質情報システムの実際

5.1 システム構築の基本方針

過去2回同様のシステムが企画されながら実現に至らなかった経緯がある。当社の場合、各担当部署におけるパソコンでの品質情報処理が比較的進んでいた。しかし各部署毎に情報のコード体系、レコード長、属性等、情報の持ち方が異なる状況にあった。過去の企画では、これら各部署に分散していた情報をコード体系等すべてを統一してからホストコンピュータに投入しようと考えたため各サブシステムの負担が大きくなり過ぎ、実現に至らなかった経緯がある。このため今回のシステム構築に当っては、既存サブシステムの変更は極力最小限にとどめることとした。このため内容的に不十分なサブシステムもあるが、第1次計画では、まずこれらの定着に主眼を置くことにした。システム構築に当たっての基本方針は下記の通りで

ある。

- 1) システムは将来の拡張性を充分考慮する
- 2) 各サブシステムの変更は極力最小限とする
- 3) ホストコンピュータにRDBを導入し、情報の一元化を図る
- 4) ホストコンピュータ、パソコン間を接続し、品質情報をユーザに開放する
- 5) 最終的な情報の処理、加工はユーザに委ねる

5.2 システム概要

システムの概念図を図-12に示す。部品受入検査⇒製造工程⇒検査工程⇒納入先品質情報⇒市場品質情報、という当社製品の流れに沿い、それぞれのステップで担当部署が品質情報の収集を行っている。各担当部署は、これらの情報をパソコンレベルで処理し、更にホストコンピュータ上のRDBへ転送する。ホストコンピュータに投入された情報は、担当部署以外のどの端末からでも、常時（深夜を除く）アクセス可能となっている。以下、各事項毎に概略を紹介する。

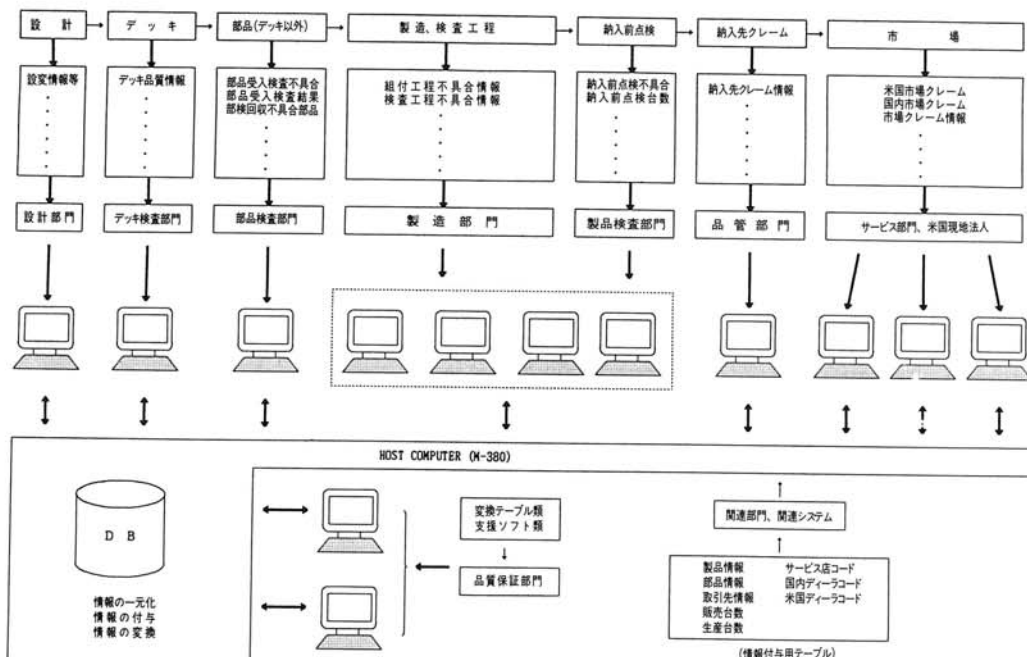
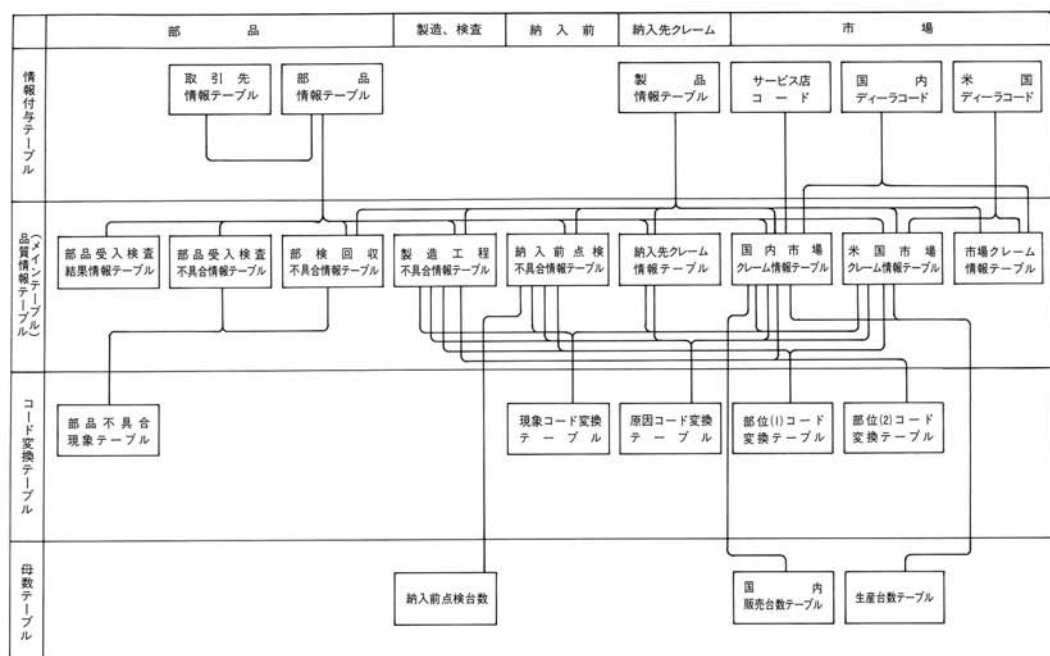


図-12 品質情報システム概念図

Fig. 12 Scheme of our quality information system.



(注、テーブル間の線は、テーブルの関連を示す)

図-13 品質情報システム テーブル関連図

Fig. 13 Diagram of related data tables.

1) テーブル

第1次計画におけるテーブルは、部品品質情報関連で3テーブル、製造検査工程で2テーブル、納入先クレーム1テーブル、市場クレーム関連で3テーブルの計9テーブルを品質情報（メイン）テーブルとした。この他に情報付与用、コード変換用、母数用の周辺テーブル14を作成した。全テーブルと、その関連を図-13に示す。（63年8月現在）

2) ハードウェア

主要ハードウェアとしては、ホストコンピュータにM-380S、端末としてパソコン（FACOM-9450系、FM-R系）13台をホストコンピュータに接続した。ホスト専用端末ではなくパソコンを使用したのは、最終的なデータの処理加工は、ユーザが使い慣れたパソコンをそのまま使用できるよう配慮したためである。

製造、検査工程等のサブシステムにおいては、

現場での情報入力の容易性、省力化のため、タッチパネルの採用、入力画面の作成等の配慮をした。

3) ホストコンピュータ用ソフトウェア

主要ソフトとして、AIM/RDB^{脚注4)}および、端末からの対話型システムを可能とするため、INTERACT^{脚注5)}を用いた。ソフトウェア環境を図-14に示す。

4) パソコン用ソフトウェア

テーブル転送用にF6650エミュレータ、EPODUET（以上FACOM-9450系）または、F6680エミュレータ、FMPORT、通信タスクモニタ（以上FM-R系）を用いた。

データ処理用ソフトとしては、各ユーザが従来から使用している使い慣れたEPOACE等のソ

脚注 4) AIM/RDB: advanced information manager/relational data base (富士通)

脚注 5) INTERACT: interactive integrated system (対話型統合データ処理システム 富士通)

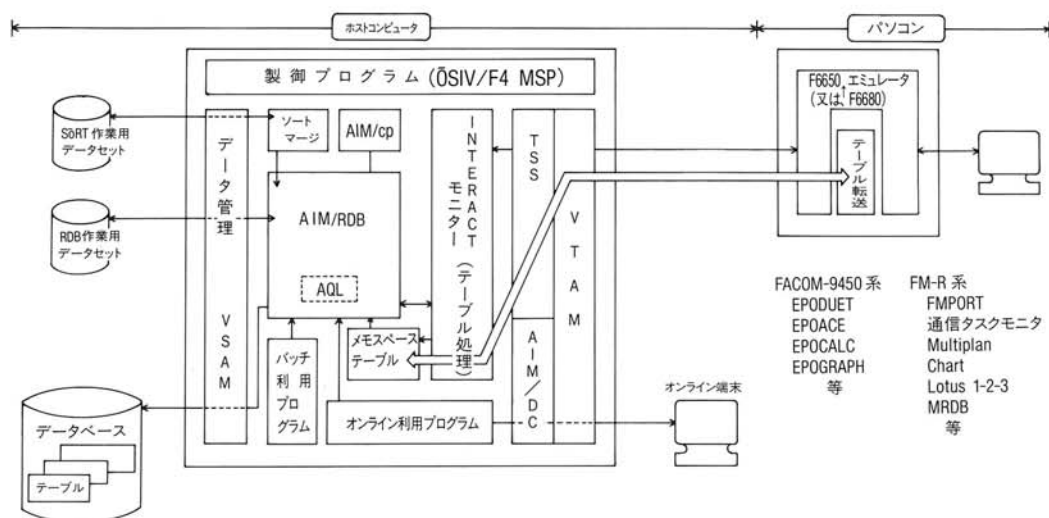


図-14 品質情報システムソフトウェア環境

Fig. 14 Software diagram of our system.

フトをそのまま使用することとした。

5.3 サブシステムとテーブル

部品検査部門、製造部門、製品検査部門、品質管理部門、国内サービス部門、海外サービス部門をサブシステムとし、それぞれの担当情報をホス

トコンピュータに入力した。これらテーブル（品質情報（メイン）テーブル）の概要を表-3に示す。

5.4 周辺情報テーブル

周辺情報テーブルは、品質情報（メイン）テーブルの機能を十分に発揮させるためのテーブルで、

表-3 品質情報（メイン）テーブル

テーブル名		内 容	カラム数	レコード長 (バイト)
1	部品受入検査結果 情報テーブル	社外から購入する部品の受入検査結果を表すテーブル	30	91
2	部品受入検査不具合 情報テーブル	上記1のテーブルから、不合格部品のレコードを抜き出したテーブル	8	57
3	回収不具合部品 情報テーブル	後工程で発生した不具合部品を回収調査し、その結果を表すテーブル	13	113
4	製造工程不具合 情報テーブル	組立製造工程および出荷検査工程において不具合になった製品の、解析結果を表すテーブル	24	214
5	納入前点検不具合 情報テーブル	納入前点検工程で不具合になった製品の、解析結果を表すテーブル	26	291
6	納入先クレーム 情報テーブル	納入先において、不具合が発生した製品につき、その解析結果を表すテーブル	20	169
7	米国市場クレーム 情報テーブル	米国市場でクレームになった製品の解析結果を表すテーブル	31	315
8	国内市場クレーム 情報テーブル	国内市場でクレームになった製品の解析結果を表すテーブル	31	273
9	市場クレーム 情報テーブル	市場(全世界)で不具合になった製品の情報を、当社製品が装着されていた車両を中心に見た市場クレームのテーブル	37	227

表-4 情報付与用テーブル

テーブル名		内 容	カラム数	レコード長 (バイト)
1	取引先 情報テーブル	取引先コードをキーとし、取引先名等の情報を得るための テーブル	3	51
2	部品 情報テーブル	部品仕様をキーとし、該当部品の検査方法、検査水準など を引き当てるためのテーブル	16	60
3	製品 情報テーブル	製品型名をキーとし、該当製品の機能、向先、仕様などを 引き当てるためのテーブル	10	86
4	サービス店コード テーブル	市場クレーム関係のテーブルから、サービス店コードをキー とし、サービス店の情報を得るためのテーブル	3	39
5	国内ディーラコード テーブル	市場クレーム関係のテーブルからディーラコードをキーとし、 ディーラ情報（国内）を得るためのテーブル	3	65
6	米国ディーラコード テーブル	市場クレーム関係のテーブルからディーラコードをキーとし、 ディーラ情報（米国）を得るためのテーブル	7	112

情報付与用テーブル、コード変換用テーブル、各種不具合率を算出するための母数テーブルからなる。

5.4.1 情報付与用テーブル

情報付与用テーブルは、テーブル内のあるカラムをキーとして、関連情報を連鎖的に引き出すためのテーブルである。これは後述（5.5節）するように事前に情報を付与するものではなく、ユーザが必要時に、自ら情報を付与するためのテーブルである。情報付与用テーブルの概要を表-4に示す。

5.4.2 コード変換用テーブル

基本方針で述べたように、各サブシステムは原則的に、現象・原因等のコードは独自のものを使用している。このため複数のサブシステムを統合して見る必要が生じた場合には、各サブシステムの独自コードを統合コードに変換し、同一のコードで見る必要がある。コード変換テーブルは、この変換を可能にするとともに、コード体系の辞書としても使用できるよう配慮した。コード変換用テーブルの概要を表-5に示す。

5.4.3 母数テーブル

各種不良率を算出するため、3種の母数テーブルを用意した。母数テーブルの概要を表-6に示す。

表-5 コード変換用テーブル

テーブル名		内 容	カラム数	レコード長 (バイト)
1	部品不具合現象 コードテーブル	部品関連情報テーブルから、不具合現象コードをキーとし、 その内容を引き当てるテーブル	3	63
2	現象コード 変換テーブル	製品関連情報テーブルから、現象コードをキーとし、統合 現象コードに変換するためのテーブル	10	167
3	原因コード 変換テーブル	製品関連情報テーブルから、原因コードをキーとし、統合 原因コードに変換するためのテーブル	10	185
4	部位(1)コード 変換テーブル	製品内の不具合発生部位を、大分類コードで表す時、その 内容を引き当てるためのテーブル	10	75
5	部位(2)コード 変換テーブル	製品内の不具合発生部位を、中分類コードで表す時、その 内容を引き当てるためのテーブル。（なお、不具合発生部 位の小分類については、コードではなく文字で表す）	6	99

表-6 母 数 テ ー ブ ル

テ ー ブ ル 名		内 容	カラム数	レコード長 (バイト)
1	納入前点検 台数テーブル	納入前点検実施台数を、点検日毎、製品型名毎に表すテーブル	6	6 5
2	販売台数テーブル	月毎、営業所毎の販売台数を表すテーブル	3	3 2
3	生産台数テーブル	製品型名毎、生産月毎の、生産台数を表すテーブル	5	3 4

5. 5 情報の変換、情報の付与

前述した各種テーブル内の情報は、サブシステムからRDBのテーブルにデータを投入する時点で、ホストコンピュータにより、情報の変換（レイアウトの変更を含む）、情報の付与を行っている。情報の変換、付与の概要を次に述べる。

5. 5. 1 情報の変換

基本方針で示した通り、サブシステムの変更は極力最小限にとどめ、サブシステムの独自性を失わないよう配慮した。一方ホストコンピュータ上では、データの形式が統一されていることが望ましいのは言うまでもない。従ってデータを変換しても、その精度低下を招く恐れのないものに関しては、ホストコンピュータにおいて変換をかけ、情報形式の統一を図った。変換の例を次に示す。

例-1 レイアウトの変換

サブシステム（パソコン）の情報は、ホストコンピュータのRDB上のテーブルに投入する時点で、標準として設定したレイアウトに、すべて並べ換えた。

例-2 年月日

テーブル上には年月日カラムが6種あるが、各サブシステム毎に、西暦を使用しているもの、元号を使用しているものがあり、また米国関連の情報は、 $\cdot\cdot\cdot$ の順に並んでいる。このためこれらは全て、年月日の順に並べ換

え、年については変換により西暦下2桁に統一した。これは、端末として使用するパソコンでの期間計算を容易にするための配慮でもある。

例-3 製品型名の変換

製品型名として各サブシステムは、正式型名、集約型名、OEM型名のどれかを使用している。

従って、該当サブシステムで使用している型名から、他の2つの型名に変換をかけ、かつ全ての型名をテーブル上に持ち、どの型名からも検索できるように配慮した。

5. 5. 2 情報の付与

各テーブルの情報に対し、前記した変換の他に、情報付与の操作も行っている。これは、品質情報システム以外の生産情報や営業情報等から、品質情報システムに有用な情報を取り込むためである。

情報付与の例を下記する。

例-1 製造年月日の付与

市場クレーム等で修理解析を行った製品の情報1件1件に対し、該当シリアルナンバーをキーとして、生産ロットマスタから生産年月（日）を付与している。

例-2 向先情報の付与

製品型名から、該当製品の出荷先をコードに

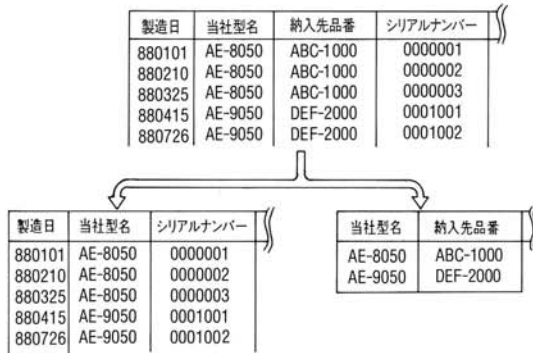


図-15 テーブルの分割

Fig. 15 Separate control of a table.

て付与している。

例-3 製造場所情報の付与

製品型名、生産年月(日)から該当製品の製造場所をコードで付与している。

以上の例で述べたような情報付与は、一般的には付与したい情報を格納するテーブルを別途作成の上、必要時にJOIN(結合)し、その時点で利用者自身が情報の付与を行うべきである。ある情報が複数のテーブルに重複して存在する時には、その重複部分を1つの別テーブルに分割すべきである(図-15)。何故ならば、テーブル分割によ

り資源の節約が図れ、かつ、情報の更新時点においても1つのテーブルのみ変更すれば良く、変更漏れ等の不具合の発生を防止できるからである。

しかし、当システムでは完全なテーブル分割は行わず、冗長さのあるテーブルを許している。これは、現状のRDBには処理速度の問題があり、情報の利用時点で、その都度情報付与を行うのは、時間が掛かり過ぎ現実的でないからである。

従って使用頻度の高い情報については、敢て冗長性を許し、各テーブルに事前に情報付与を行っているのである。

5. 6 運 用

運用につき、その概要を述べる。

5. 6. 1 RDBの利用形態

RDBの利用形態は多様であるが、主要なものは下記の通りである。

- 1) INTERACT フルスクリーンモードでの、メニュー画面からの利用(図-16)
- 2) INTERACT ラインモードでの、AQLコマンドの直接押下による利用
- 3) データをホストコンピュータからパソコンへテーブル転送し、パソコンで処理を行う利

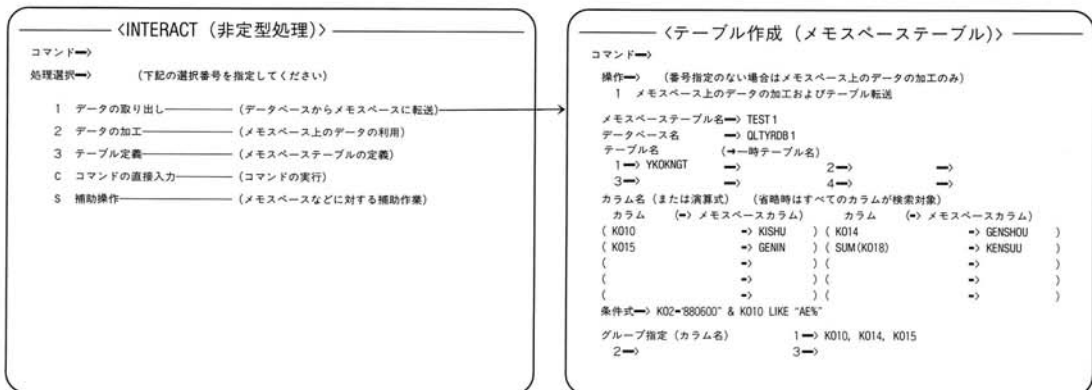


図-16 INTERACTメニュー画面によるRDB操作例

Fig. 16 Data processing by INTERACT menu (CRT display).

《テーブル転送》

コマンド⇒ (下記の選択番号を指定してください)

1 テーブル受信 (メモスペースから端末に転送する)

2 テーブル送信 (端末からメモスペースに転送する)

メモスペーステーブル名⇒ TEST 1

メモスペーステーブルの置換え⇒ (Y: する, N又は省略: しない)
(テーブル送信時のみ有効である)

《テーブルの受信》 (26/46) V25/L01

ホストよりテーブルを受信します。

テーブル名 : TEST 1
テーブルの行数: 000000

・受信するテーブルの形式を指定してください。⇒2

1. EPOC 形式
2. バインディング形式
3. テキスト形式
4. COBOL データ形式

・名前を変更する場合はテーブル名を指定してください。⇒TEST100

・テーブルのユニット番号を指定してください。⇒13

・受信するテーブルは新規ですか、置き換えますか? (1: 新規 2: 置き換え) ⇒1

・受信したテーブルをホストに送ることはありますか? (Y: はい, N: いいえ) ⇒

改行: 次のパラメータ指定 入力: パラメータの指定完了 リセット: 転送キャンセル

図-17 テーブル転送画面

Fig. 17 CRT display for table transmission.

用 (図-17)

4) コマンドプロシジャーによる定型処理の実行 (図-18)

5) COBOL, FORTRAN等による応用プログラムからの利用

当社において、品質情報システムにRDBマネジメントシステムを用いたのは、既述の通り、情報の集約加工の観点が一貫でない非定型処理が多く、かつ情報をユーザに解放し、その加工をユーザに委ねるためである。従ってその利用形態は非定型の処理が中心となる。具体的には、まずINTERACT のメニュー画面または、コマンドの

直接押下により、ある程度の情報処理をホストコンピュータに実行させる。さらに実行結果をテーブル転送によりパソコンへ取り込み、パソコンで最終処理をするという利用形態 (前記 1) 2) と 3) の組み合わせ) である。また、定期的な出力の必要が生じたものに対しては、コマンドプロシジャーや独自メニューを作成し、定型処理を可能とした。(前記 4)

5. 6. 2 品質情報の更新

サブシステムパソコン上の情報は、テーブル転送により、ホストコンピュータINTERACT 配下のメモスペースへ送られ、一旦順編成ファイルで

《国内市場クレーム情報の検索》

処理選択⇒ (下記の選択番号を指定してください)

1 管轄営業所別クレーム発生状況表 (集計期間指定)

2 管轄営業所・サービス店別クレーム発生状況表 (集計期間指定)

3 製造場所別クレーム発生状況表 (集計期間指定)

4 部門別クレーム発生状況表 (集計期間指定)

5 機能別クレーム発生状況表 (集計期間指定)

6 現象・原因マトリクス (年月とセット当社集約型名指定)

7 国内市場クレーム情報の検索 (項目選択)

《国内市場クレーム情報の検索》

(国内市場クレーム情報の検索)

☆選択項目のみにSを入力して下さい

選⇒ K01 情報識別 CD	選⇒ S K02 集計年月日
選⇒ K03 追番	選⇒ K04 生産年月
選⇒ S K05 登録年月日	選⇒ K06 不具合セット受付年月日
選⇒ K07 修理・解析年月日	選⇒ K08 向先コード
選⇒ S K09 セット当社型名	選⇒ S K10 セット当社集約型名
選⇒ S K11 納入先品番	選⇒ S K12 シリアル NO
選⇒ K13 生産場所	選⇒ K14 ユーザー不具合現象 CD
選⇒ K15 不具合原因 CD	選⇒ K16 解析後不具合部位 CD
選⇒ K17 主原因部品	選⇒ S K18 不良数
選⇒ K19 ディーラー CD	選⇒ S K20 車両フレーム NO
選⇒ K21 車両形式	選⇒ K22 サービス店(S/S) CD
選⇒ K23 修理 NO	選⇒ K24 サービス区分
選⇒ K25 追記年月日	選⇒ K26 追記者

検索条件⇒ K02: "880100" & K02<"880600"

追記条件⇒ K010

MST 名⇒ TEST 1

図-18 独自メニュー画面による定型処理の例

Fig. 18 Our patternized processing menu.

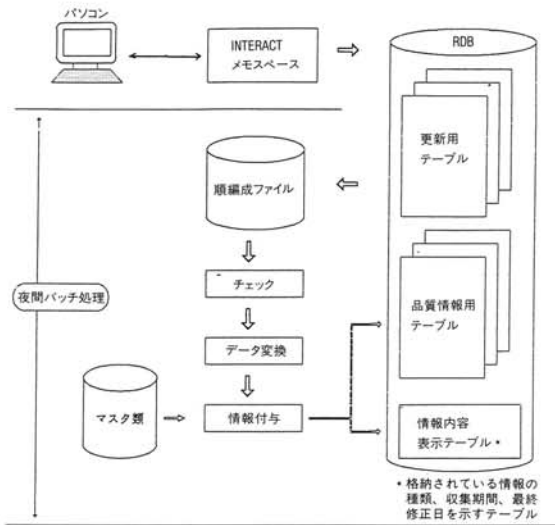


図-19 データベース更新フロー概要
Fig. 19 Outline of data base renewal.

経て、情報の変換・付与を行い、夜間バッチ処理によりテーブルを更新する。バッチ処理を行っている理由は、1) データベースの破壊を防ぐこと、2) 事前にデータのチェックを行うこと、

3) 情報の変換、付与を行うこと等のためである。このため更新は、リアルタイムではなく、最大24時間のタイムラグが発生するが、現時点では実用上問題は無いと考えている。更新のフローを図-19に示す。

5. 6. 3 端末での情報処理

端末での処理は5.6.1 RDBの利用形態で述べた通りであり、処理は原則的にユーザに委ねることとした(図-20)。なおパソコンの画面分割を利用し、AQLコマンドの解説、テーブル名・カラム名の説明等のオフラインヘルプの機能を設け、利用者の便宜を図った。これを図-21に示す。

5. 6. 4 機密保護

品質情報に関しては、社内的には機密事項は存在しないが、データベース破壊の恐れがあるため、端末使用開始時およびデータベースアクセス時に、それぞれパスワードを設定し、利用端末、利用者の制限を行っている。

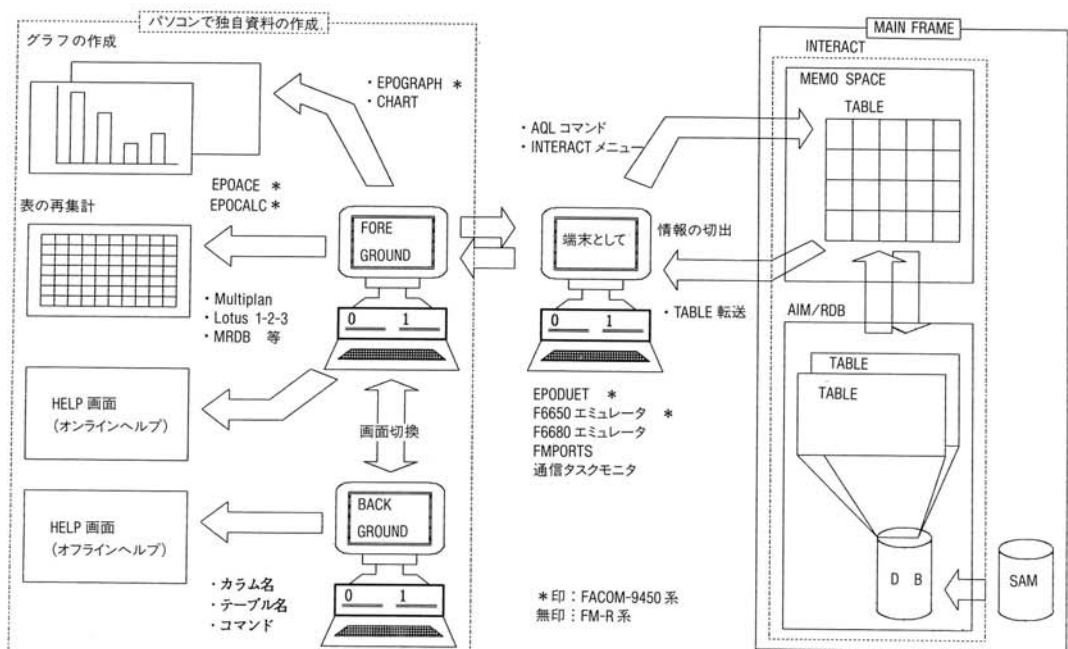


図-20 品質情報システムデータ処理概念図
Fig. 20 Data processing of quality information.

RDB 操作用語									
1 開始	5 登録	9 計算	13 変更	17 空き領域					
2 使用宣言	6 読み	10 分類	14 中断	18 属性表示					
3 使用終了宣言	7 表編集	11 削除	15 画面操作	19 ヘルプ					
4 検索	8 表作成	12 転送	16 表(項目)名一覧	20 終了					

1 上記の中よりこれから行ないたい仕事を番号で入力して下さい 4

1 SELECT 2 AND 3 OR 4 BACK 5

6 7 8 9 10

2 上記の中より知りたいコマンドを番号で入力して下さい 1

コマンド: SELECT 種別: AQL コマンド 省略形: SEL

意味: 指定した条件を満たす行をテーブルから検索し画面に表示する

有効範囲: AQL モード (既に START AQL が入力されている時のみ有効)

補足: このコマンドについての詳細情報を知りたい場合、RDB のコマンド入力行に H SEL を入力して下さい 右画面: 3 →

全取消し
項目取消
終了

(コマンドの説明例)

右画面: 1
米国市場クレーム情報 YUSANGT
表示ページ 2P

3
前ページ: 2
次ページ: 3

カラム名	カラム見出し	属性	内 容
US14	シリアル	X (8)	シリアル NO
US15	生場	X	生産場所
US16	現象	X (2)	ユーザ不具合現象コード
US17	ユーザ不具合現象コメント	X (40)	ユーザ不具合現象コメント
US18	部 1	X (1)	ユーザからの (AM 部、カセット
↑	↑	↑	部等) 不具合部位
US19	テ現象	X (2)	TEN (S/S) 確認不具合現象コード
US20	不具合原因コメント	X (64)	不具合原因コメント
US21	原因	X (2)	不具合原因コード
US22	部 1	X	(S/S) 確認の (AM 部、カセット
↑	↑	↑	部等) 不具合部位
US23	主原因部品	X (24)	主原因部品
US24	不良数	9 (7)	不良数
US25	ディーラ	X (10)	ディーラコード

右画面を見るなら、*1* を押して下さい。

全取消し
終了

(カラム、属性、内容の説明例)

図-21 オフラインヘルプ画面の例

Fig. 21 Examples of off-line help (CRT display).

5. 6. 5 情報の保存期間、容量

情報は原則として10年間保存することとした。このうち1～3年分(テーブル毎に異なる)の情報をDASD上に常駐させることとした。常駐させる情報量は現時点では350M バイト分を確保しているが、必要に応じて逐次増加させる予定である。

6. 品質情報システムの限界と展望

6. 1 品質改善用情報とマネジメント情報

今回構築した品質情報システムでは、原始データを直接RDBに蓄積しているため、色々々観点からデータの集計加工(地域別の品質、使用期間と部品不良の関係、車両走行距離と当社製品の不具合発生との関係、等々)が行える。これらを集約加工したものはマネジメント情報として有益であり、また試行錯誤的解析を行ったものは、品質改善のために、大いに役立っている。(出力例を図-22に示す。)

しかしながら、品質改善のための情報としては不足を感じる面がある。精密な品質改善を実施するに当り必要な情報とは何かを吟味すると、不具合が発生した現物そのものが一番有益な情報であることが判る。例えば、はんだ付が外れた不具合では、文字(またはコード)でいくら表現しても、現物から得られるはんだの艶、破断の状態等の多くの情報には及ばない。従って品質改善用の情報としては、EDP化された文字情報を、どこまで不具合現品そのものが持つ情報に近づけ得るかということが課題である。

6. 2 文字情報の限界

当社品質情報システムは、RDB上の文字情報であり、6.1で述べたように不具合現品そのものが持つ情報量との間には大きなギャップがあり、これが品質改善用としての情報のネックになることがある。このギャップを縮めるため、今後はイメージ情報のデータベース化等の検討が必要である。

6. 3 コード統合による精度低下と、その防止

情報精度低下の恐れが無いものについては、テーブルヘデータを投入する前に変換をかけ、デー

脚注 5) DASD: direct access storage device(直接アクセス記憶装置)

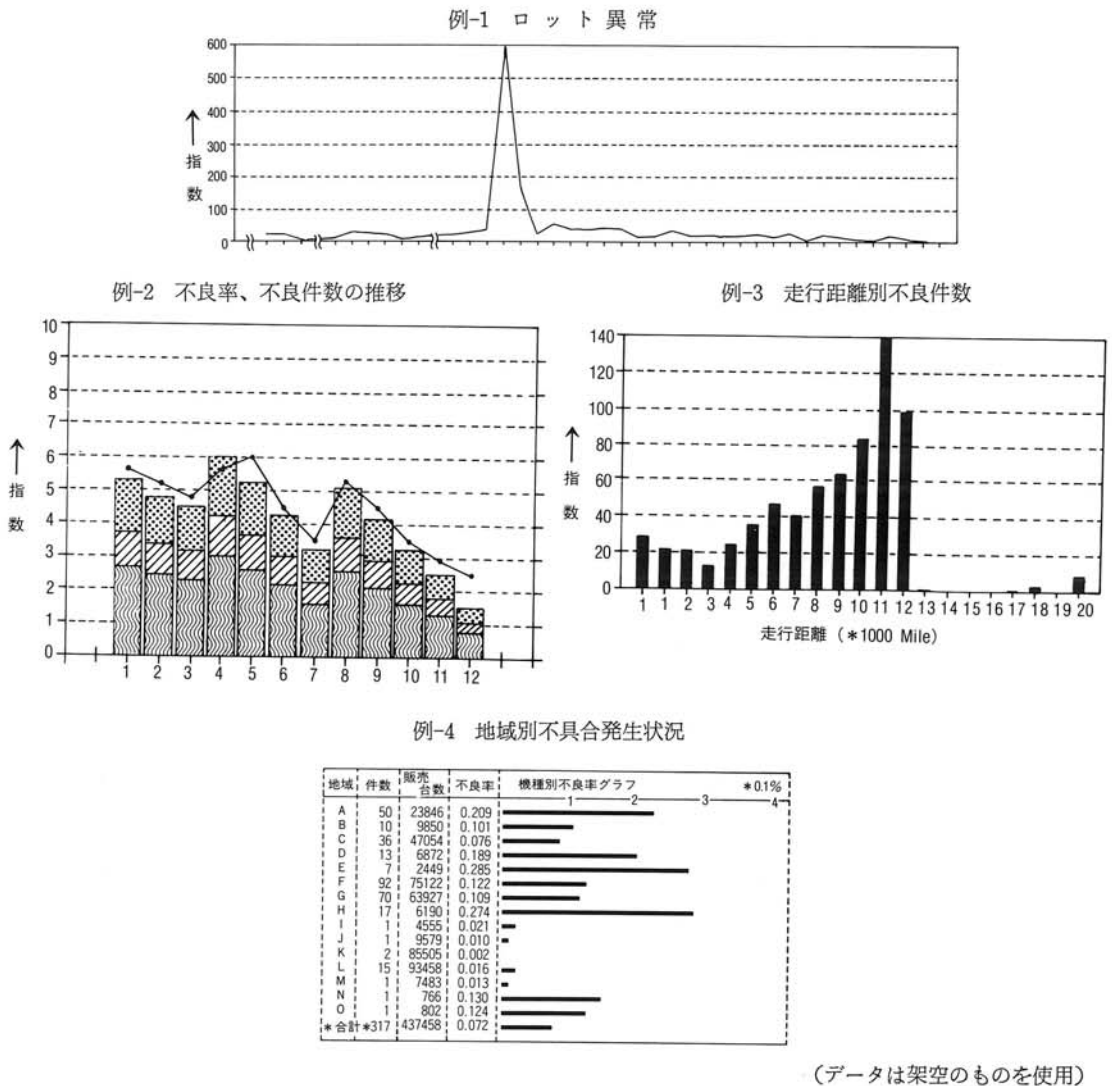


図-22 品質情報出力例

Fig. 22 Examples of output chart.

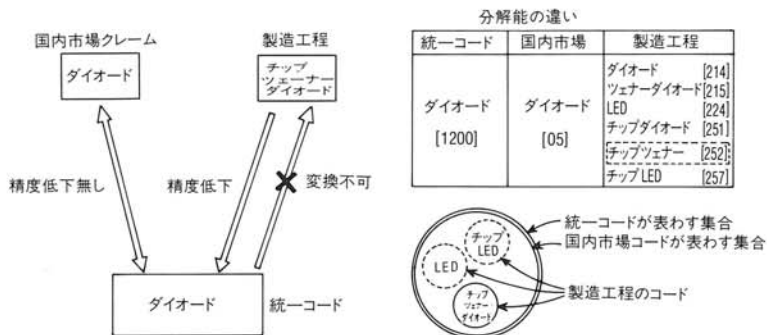
データの統一を図ったことは、5.5.1で述べた。しかし現象コード、原因コード、不具合発生部位コード等は、ホストコンピュータ上で統一してしまうと精度低下が発生する。このため、各サブシステム設定のコードをそのままRDBへ投入し、精度低下が発生しないよう配慮した。なお若干の精度低下を招いても、コードの統一が必要な場合には、コード変換テーブルにより、これを可能とした。

コード統合による精度低下の例（不具合発生部

位での例）を図-23に示す。不具合がチップツェナーダイオードの部位に発生した場合、製造関係のサブシステムでは、コード252（チップツェナーダイオード）で表わされ、国内市場関係のサブシステムでは、コード05（ダイオード）で表わされる。これらを統合するため、統合コードに変換すると、コード1200（ダイオード）で表わされる。

統一コードと、変換前のサブシステムのコードを比較すると、国内市場については、精度に変化

例-1 コードの分解能に差がある場合（不具合部位コードでの例）
不具合がツェナーダイオードに発生した場合



例-2 コードの定義する集合が異なる場合（不具合現象コードの例）

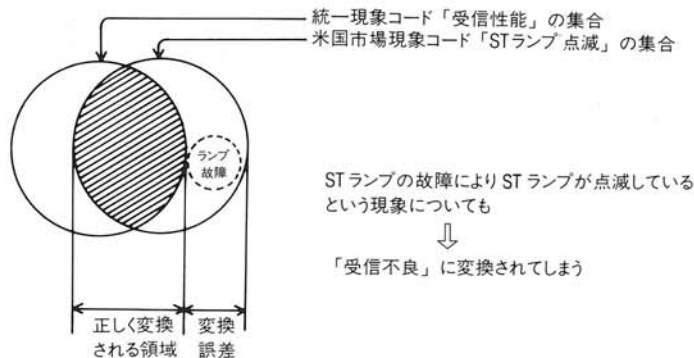


図-23 コード変換により発生する精度低下

Fig. 23 Code conversion and quality information.

は無いが、製造関係では明らかに精度が低下（チップツェナーダイオード⇒ダイオード）していることが判る。（図-23、例-1）

これは、各サブシステムで設定しているコードの分解能の違いにより発生する。つまり統一コードは、一番分解能の低いコードに合わせなければならないため、もともと分解能の高いサブシステムでは精度の低下が発生するのである。

また図-23、例-2のような場合には、精度の低下というよりは、間違った情報を発生してしまう可能性を含んでいるのである。

以上のように、当システムでは統一コードを作成し、ホストコンピュータ上でのコード統一を可能としたが、基本方針の通りサブシステムのレベ

ルまでは普及させていない。将来的には、統一コード（小分類…分解能を良くしたもの）が全サブシステムのコード統一の指針となることを希望するものである。

7. む す び

関連部署に分散していた品質関連の情報を、ホストコンピュータ上のRDBに一元化し、品質情報をユーザに解放することができた。またRDBマネジメントシステムを用いたことにより、ユーザ側からの試行錯誤的検索も可能となり、品質改善用情報として、またマネジメント用情報として社内でも好評を得ている。

当システムは、ハードウェア、ソフトウェア、

情報内容自身等、更に改良拡張すべく、現在も発展途上にある。情報内容については、更にきめ細かいステップで情報を採取する計画であり、工程内の自動測定器等との接続も検討している。遠隔地の工場についても通信回線で接続の上、情報を採り込む計画であり、既に一部は着手済である。また、情報付与用、変換用として、品質情報システム以外からも多くの情報を採取しているが、品質情報の精度アップには、これら周辺情報のレベルアップも急務である。

処理速度向上のためには、ソフトウェアの改良、品質情報用コンピュータの独立専用化の検討も必要と考えている。

最後に、本稿が今後RDBマネジメントシステムを構築される方の参考となり、社内においては多くの方が品質情報システムに興味を持たれ、品質情報システムの利用者が更に増加することを希望するものである。

参考文献

- 1) 中原編：情報の検索とデータベース、電子通信学会 (1986)
- 2) 富士通編：リレーショナルデータベースの概要 FACOMファミリ会関西支部例会資料 (1988)
- 3) 林他：リレーショナルデータベース管理システム AIM/RDB の実現手法、日経エレクトロニクスNo. 310, pp171~202 (1983)
- 4) FACOMファミリ会関西支部編：FACOMリレーショナルデータベースの活用 (1988)
- 5) 富士通編：FACOM OS/4 AIM/RDBの適用
- 6) 渡辺他：リレーショナルデータベース AIM/RDB の機能と構造、FUJITSU, Vol. 34, No. 5 (1983)
- 7) 富士通編：INTERACT使用手引書
- 8) 富士通編：AIM/RDB使用手引書