製品検査値管理システム再構築における新技法等の考案

旭電化工業　株式会社

■執筆者Ｐｒｏｆｉｌｅ■

岩切俊博

1974年　旭電化工業㈱入社
1983年　工場にてシステム管理担当
　　　　製造管理，検査値管理システムの開発と運用を担当
現在　　システム部にて，主に検査値管理システム
　　　　を統括的に管理.

渡瀬浩一

1968年　旭電化工業㈱入社
　　　　化学品開発研究所
1986年　システム部システム課長
1995年　技術士（情報工学）登録
現在　　システム部長

■論文要旨■

昭和63年より各事業所でF9450 を採用して製品検査値管理システムを展開したが，ホストコンピュータと接続して，販売物流システムにドッキングして全国の倉庫で，出荷時自動添付するなどの大きなシステムに発展させることができて，本システムにより合理化効果は発揮された．然し，長期間に種々のシステムが作られ保守も対応不可能な程複雑になってしまったこと，パソコンのOSもMS-DS に世代交代し出したことから平成２年にエースステーション350 Σ(2台目からはDS/90)をサーバーとし、FMRパソコンをクライアントとするIA/FRAMEをミドルウエアとし，データベースマネージメントシステムにINFORMIX，プログラム言語はＣを採用して，製品検査値管理システムの統合化システムの開発を開始した．

開発に当たって，従来からの事業所ごとに特徴のある部分を生かし且つ全体としては統合したシステムを目指す検討を行い，パッケージ化と追加型カストマイズ方式を考え出して統一と独自性のバランスを取り開発生産性を大きく上げた．

実際のシステム開発においては，従来使用していたAPCSのBASIC ，ISAMとは大きく異なり，しかもＣ言語の生産性は非常に低いことから，これらを改善するため，設計手法，ツールの採用などの種々の工夫，考案をおこなって達成することができた．

統合化した新製品検査値管理システムは，製造～検査～物流業務の連携による合理化に成功した．

■論文目次■

１．はじめに -- 《　》
２．システム開発の経緯 -- 《　》
　２．１既存システム構築の変遷
　２．２既存システムの限界
　２．３新システム構築の基本方針
　　２．３．１検査／品質管理システムの機能強化
　　２．３．２システムのパッケージ化と追加型カスタマイズ方式
３．新システムの概要 --- 《　》
　３．１クライアント／サーバシステムの採用
　３．２システム全体構成
　３．３．ホストコンピュータ側システム
　３．４ネットワーク構成
４．新システム開発・運用での問題点 ------------------------------ 《　》
　４．１プログラム開発の問題点
　　４．１．１プログラム設計法の考案
　　４．１．２ＹＰＳの導入
　　４．１．３サーバー連携システムの考案
　４．２システム運用の問題
　　４．２．１処理レスポンスの問題
　　４．２．２プログラム維持の工夫
５．新システムの効果 --- 《　》
　５．１全社統合製品検査値システムの実現
　５．２製造～検査～物流業務の連携合理化
　５．３検査業務自体の合理化
６．まとめ -- 《　》

■図表一覧■

図１．標準パッケージとカスタマイズ ------------------------------ 《　》
図２．検査値管理のプログラム体系 ------------------------------- 《　》
図３．ネットワーク構成図（検査値関連） ---------------------------- 《　》
図４．検査計画のＪＳＤ --------------------------------------- 《　》
図５．サーバ連携システム図 ------------------------------------ 《　》
図６．検査値管理のチェーン図 ----------------------------------- 《　》
図７．三重工場での新旧対比図 ----------------------------------- 《　》

１．はじめに

弊社は，1917年に，食塩水の電解法によるカセイソーダ，塩素，水素の製造を開始し，現在は，各種の石油化学原料を有効利用した無機，油脂，有機の各種中間製品から電材，潤滑油などのファインケミカル製品及び，動植物油脂原料を高度に利用したマーガリン，ショートニング等加工油脂，冷凍，無菌化等の技術による加工食品の生産を行っている．工場は，相馬，鹿島，千葉，富士，三重，明石，境港，この外に，研究所が尾久，浦和，久喜にある．

本小文は，これら各々の事業所で個々に進めてきた製品検査，品質管理のシステム化を統合的に再構築した際に考案，導入した手法，技法についてまとめたものである．

２．システム開発の経緯

２．１　既存システム構築の変遷

最初の製品検査値管理システムは，昭和57年に，千葉工場で”F9450 パーソナルターミナル”（外部記憶装置としては 5in／320 KBフロッピー装置 4デッキと 8in／１MB装置１デッキのシステム構成）を使用して開発し稼働を開始したものである．本システムの業務内容は，分析データ整理，管理図の作成，分析データを要求してきた顧客に対して分析表を発行するもので，データ項目の主体は数値であった．

昭和58年からは，当工場構内に独自に構築したパソコンLAN 上でデータの共有化を図った．その後，各工場で同様のシステムをそれぞれの工場に合わせて開発し稼働させた．

昭和63年，一部工場の製品検査値管理システムのパソコンとホストコンピュータを回線接続して検査値管理に関わるデータ管理システム及び，全国の倉庫に設置されている販売物流システムのホストオンライン端末より納入先向試験成績表を出力するシステムを開発し，一部の倉庫で倉出指図書に自動添付することができるようになった．

２．２　既存システムの限界

この間，F9450 も機能／性能が強化され，本システムの中核として大きな役割を果した．しかし，各工場でのF9450 及び構内LAN の活用が進むにつれて，検査項目の拡大，品質管理レベル向上，帳表多様化などの要求に個々に対応した結果，パソコンおよびパソコンLAN では，処理内容，処理速度の限界が出てきた．

更にコンピュータを積極的に活用して，検査作業自体の能率アップ，製造管理システムと連携させた検査業務スケジューリングのシステム化も要求されるようになった．

２．３　新システム構築の方針

こうしたことから，平成２年より現状システム分析を行い，次の方針で再構築の検討に入った．

２．３．１　検査値／品質管理システムの機能強化

従来は，検査データ規格チェック，管理図等の数値計算と図表作成が主目的であった．今回のシステム化では機能強化として，工場内の製造管理システム及び販売物流システムとの連携の他に，個々に対応して煩雑になってしまった検査表発行システムを抜本的に見直して，標準試験成績表発行機能と次の⑵～⑸に示す四つの新機能に整理することにした．

⑴検査／倉入業務の合理化

①製造課員は，手元のパソコンから検査依頼を入力することで，検査担当者に連絡でき

るようにする．

②検査担当者は，手元のパソコンで検査作業スケジュールを立てられるようにする．

③製品倉入／出荷業務の担当者は，検査合格情報に基づいて，販売物流システムに倉入入力するようにする．

⑵評価語方式の全面採用

既存システムの開発当初は，規格，分析結果値は数値が主体で，評価語方式を採用したものはあまりなかったが，その後，増加し，現在は100 項目を越えるようになってきて，既存システムでは個々に煩雑な対応をしており，この整理と改善は，今回のシステム化の大きな目標であった．その一例を次に示す．

・言葉で表現する例　：「適合」，「検出せず」，「合格」，「外観透明」等々
・ランキングで示す例：「Ａ」，「Ｂ」，「Ｃ」，「Ｄ」の４ランク，
　　　　　　　　　　　「５」，「４」，「３」，「２」，「１」の５ランク等々
・文章の例　　　　　：ＪＩＳ・・・検査法で適合

⑶顧客のＩＳＯ認定取得に伴う対応

顧客側でＩＳＯ認定を受ける際に，顧客より弊社に対して，原材料（弊社製品）の品質データに関して，納入規格に合格していることを証明する書類（試験成績表）に「検査責任者氏名を記載する」など種々要請された項目について，顧客各社の要請内容を整理して，対応することにした．

⑷特殊検査表出力システムの開発

「今回納入ロット」の試験結果だけでなく，「納入した最新複数ロットの試験結果のグラフを添付する」などの履歴情報も要請されるようになったことなどのために，付随サービスを行うシステムも開発することにした．

⑸英文検査表発行システムの開発

英文検査表は，従来タイプ印刷又はワープロ印刷で，手間が掛かり，ミスが多かった。そこで，規格データ・分析データは各マスターファイルから出力し，前後の文章は自由に書けるようにデータ出力とワープロ機能を一体化することにした．

ただし，メインフレームでは英小文字が使用出来ないので，ホストコンピュータにはこの機能は持たせないことにした．

２．３．２　システムのパッケージ化と追加型カスタマイズ方式

従来のやり方は，全ての工場の検査値管理システムの基本データファイルのレコードレイアウトは統一してはいたが，依頼に対する対応方法としては個々の事業所に合わせて，F9450 系（OSはAPCS）パソコンBASIC でプログラム作成しており，ホストコンピュータについても機能別にシステム開発していたことから，結果として，ホストコンピュータは２システム，パソコンシステムのバリエーションにおいては無数に近い数に達してしまい，増加する機能追加に対し保守面で，収拾がつかなくなってしまった．

そこで第１ステップとして，クライアント／サーバーシステムからホストコンピュータまでの検査値管理システムの標準パッケージを開発する．

各事業所に適用する第２ステップでは，各事業所の事情に合わせてプログラムを改造するのではなく，追加カスタマイズする方式にして，稼働後の各種プログラムの機能増強に対応できる方式を採用することにした．この概念を図１に示す．

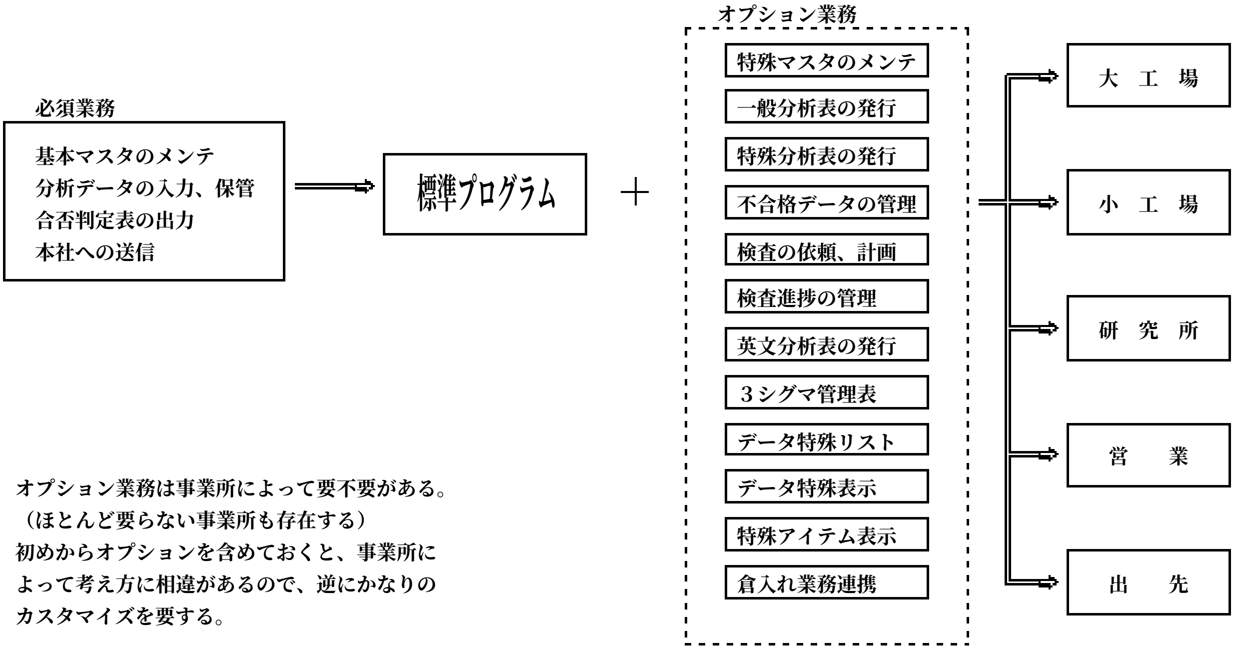

図１．標準パッケージとカスタマイズ

３． 新システムの概要

３．１．　クライアント／サーバーシステムの採用

その当時は，まだ”クライアント／サーバーシステム”という言葉はなかったと思う．しかし，これまで独自に構築してきたパソコンLAN システム（親パソコンにデータを持たせ，子パソコンにプログラム及びデータの一部も持たせる）概念を拡大していくことで種々調査検討した．

その結果，世の中の流れとしてF9450 からMS-DOSパソコンに代わり始めた時期であり，UNIXの分野で本格的な適用例が出てきたことから，DS/90 （当初はエースステーション350 Σ）を親，MS-DOSパソコンを子とし，ミドルウエアとしてIA/FRAME，データベースにINFORMIXを採用することにした．そして各パソコン端末（クライアント）には，端末用の通信ソフト、JT-IE-CAD （画面・帳表作成用のユーティリティー）で作成した定義体を置く方式を採用することを基本として，システム全体の見直しを行った．

３．２　システム全体構成

全社システムとしては，サーバーとホストコンピュータを回線接続して，販売物流システムとドッキングすることで，納入先に納入時に同時添付する試験成績表は，ホストコンピュータのオンライン処理で各倉庫に出力する．一方，特殊検査表，英文検査表，その他品質管理用のデータ処理は各サーバを使用して行うことにした．これで，これまで一部重なっていたパソコンとホストコンピュータの業務分担を明確にすることができた．

サーバ台数と配置を種々検討した結果，二ケ所の大規模事業所に関しては，担当ＳＥがおり，データ量が非常に多く，追加カスタマイズも大きくきめ細かく対応する必要があるため，現地にサーバーを設置し，それ以外の事業所は，本社システム部に設置する大型サーバを使用することにした．

３．３　ホストコンピュータ側システム

昭和62年当初の検査表発行プログラムはVSAMとCOBOL で開発しており，販売物流システムに組み込んで倉庫から出力するようにしていた．しかし，営業，工場から検査表のリクエスト発行の要請により，ADABAS-NATURALを採用してオンラインシステムを開発し対応した．その結果，「データが二重になったこと」，「プログラムメンテナンスが煩雑であること」，「VSAMとCOBOL は扱い難いこと」から，プログラム及びメンテナンスの生産性が悪いVSAMとCOBOL のシステムを廃棄し，全面的にADABAS-NATURALで統一してプログラムを開発することにした．

図２に，サーバシステムとホストコンピュータシステムのプログラム体系の説明図を示す．

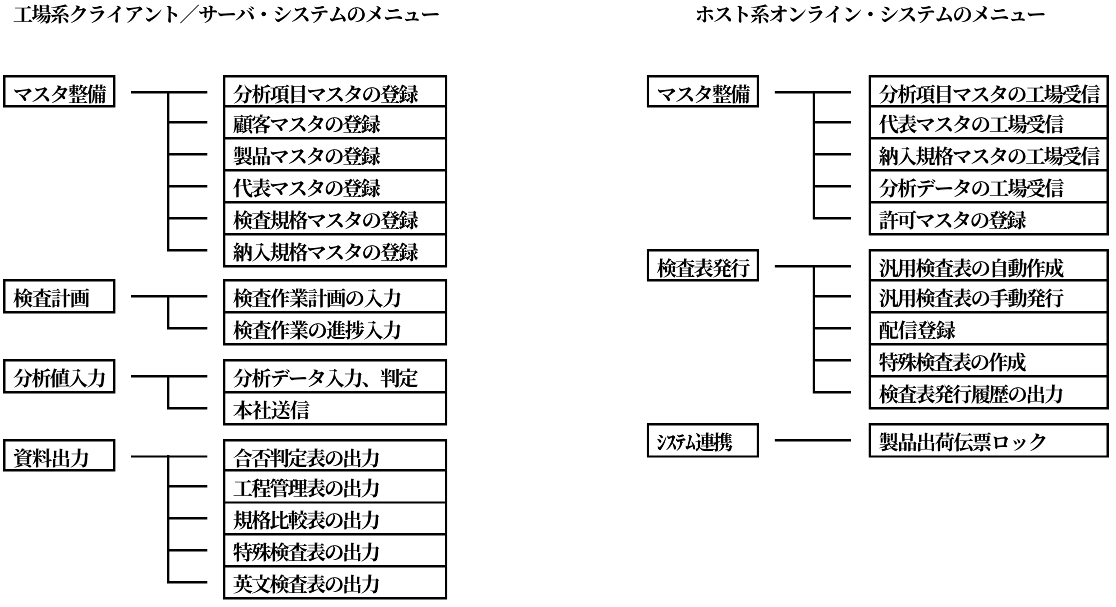

図２．検査値管理のプログラム体系

３．４　ネットワーク構成

ネットワークについては，別途構築した統合通信システム（従来のFNA に加えてトップ向け経営情報システム，電子メールシステム，社内電話を合わせたプライベートフレームリレー通信網）に全面的に乗せることにした．

図３に，検査値管理システム関係部分についてのネットワーク図を示す．

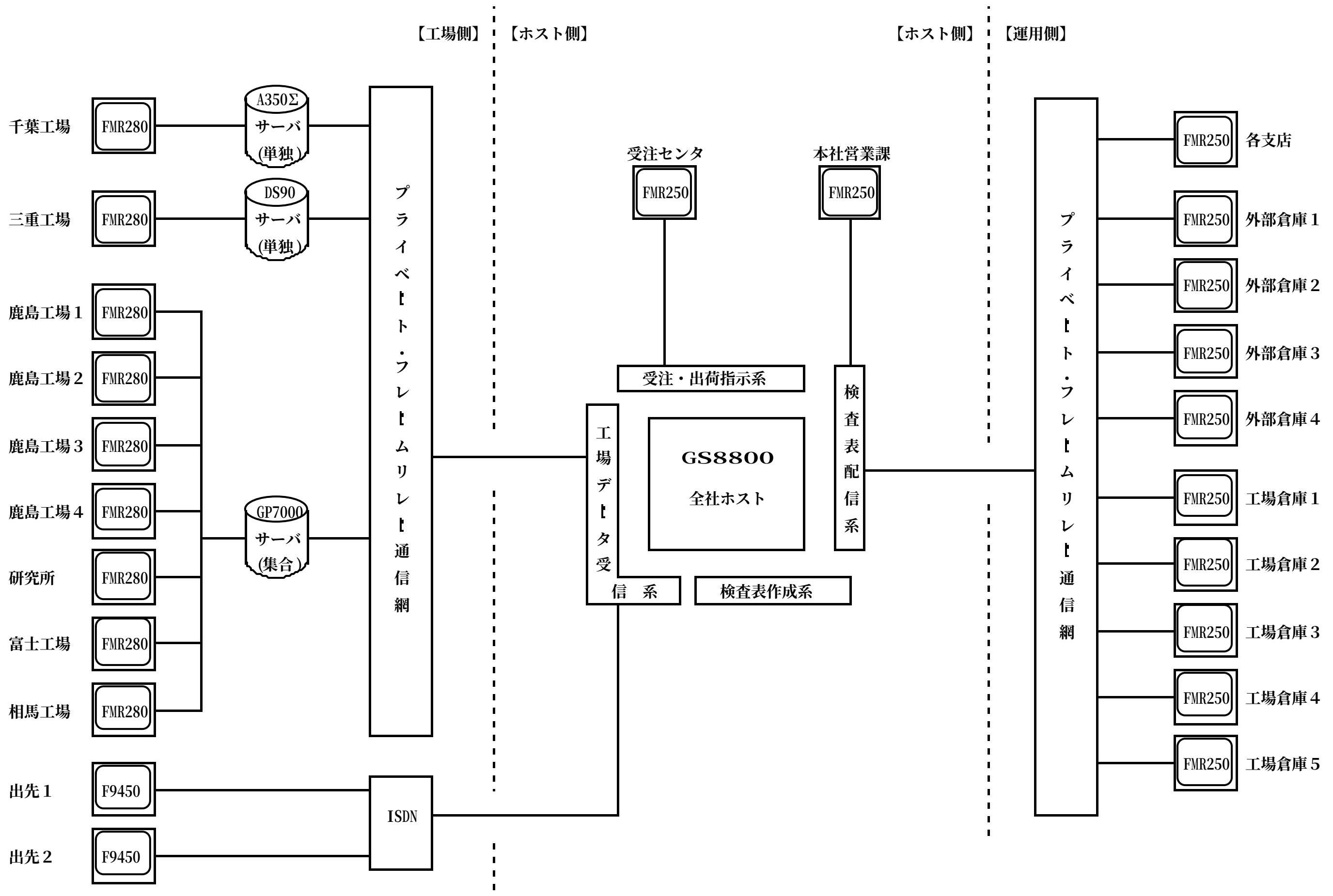

図３．ﾈｯﾄﾜｰｸ構成図（検査値関連）

４新システム開発・運用での問題点

弊社製造管理，検査値管理等のシステムは，非常に機能仕様が細かく特徴があるため，全面的に自社開発で行っている．又，従来は，APCSのBAICとISAM，SAM ファイルで，プログラムを開発してきたが，今回初めてUNIXのOS，データベース，Ｃ言語を採用したので，多少とも戸惑いは予想され，サーバ，クライアントとも，複雑なミドルウエアを搭載しており，運用面においても未知の部分が多かった．

４．１　プログラム開発の問題点

データベースは，可変長であり SQLを利用することで，データ管理は非常に容易にできたが，Ｃ言語は従来のプログラミング方式では極めて生産性が悪く，開発に予想以上の時間，計画の二倍以上かかると言う苦しみを味わった．

次に述べる「新しいプログラム設計方法の考案」，「開発ツールの導入」は，生産性を高めて効率的作業にするとともに，技術教育の面でも優れ，新人の活用という面でも大いに効力を発揮した．

４．１．１　プログラム設計法の考案

従来のテンプレート方式に対し，近年，様々な設計方法が発表されていたので，調査・検討し試行してみた結果，次のようなことが分かってきた．

①従来のテンプレート方式ではロジック追求型のボトム・アップ方式であり，どうしても「人」あるいは「場合」によって，詳細になり過ぎたり大まかになり過ぎたりしてしまい，プログラム全体を把握し難い．

②ＹＡＣ型の書き方は，縦方向に書くため，一見構造型プログラミング的で，分かり易いが，処理の詳細が横方向に書くため，

・紙面あるいは画面がすぐいっぱいになってしまう．

・詳細部分を書き過ぎると，左側の線（括弧等）が幾重にもなり，分かり難いなどの問題がある．プログラミング段階で，構造を見直すことになることが多い．

一方，Ｃ言語は関数言語であることから，「関数を作り」，「組み立て」，「まとめあげていくこと」でプログラミングを進めて行くことから，

①プログラム全体が見渡せること

②処理をブロック化し，そのブロックが更に小さいブロックに分かれるように，水平（同一フェーズ内）に処理を整理し，その上でレベルを深くしていくような手法であること

などから，マイケル・ジャクソン法を参考に，我々独自にプログラム構造作図法を考案した．これを「Job Structered Diagram法」（略称＝ＪＳＤ）と名付けた．
図４に簡単なサンプルを示す．

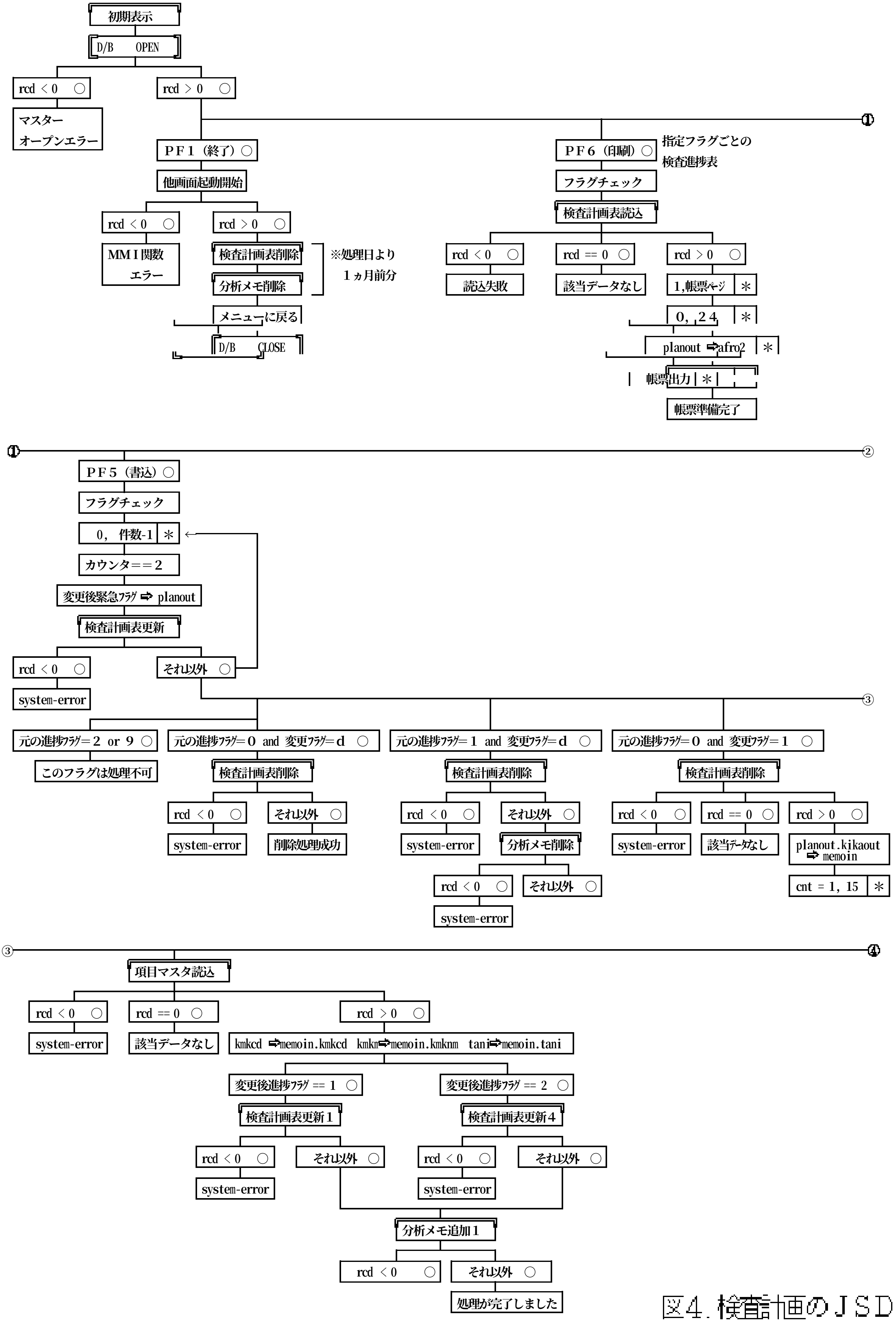
初期表示
D/B OPEN
rcd < 0 ○
rcd > 0 ○
マスター
オープンエラー
PF1（終了）○
他画面起動開始
rcd < 0 ○
rcd > 0 ○
MMI関数
エラー
検査計画表削除
分析メモ削除
※処理日より
1ヵ月前分
メニューに戻る
D/B CLOSE
PF6（印刷）○
指定フラグごとの
検査進捗表
フラグチェック
検査計画表読込
rcd < 0 ○
rcd == 0 ○
rcd > 0 ○
読込失敗
該当データなし
1,帳票ページ ＊
0, 24 ＊
planout ⇨afro2 ＊
帳票出力 ＊
帳票準備完了
PF5（書込）○
フラグチェック
0, 件数-1 ＊
カウンタ==2
変更後緊急フラグ ⇨ planout
検査計画表更新
rcd < 0 ○
それ以外 ○
system-error
元の進捗フラグ=2 or 9 ○
このフラグは処理不可
元の進捗フラグ=0 and 変更フラグ=d ○
検査計画表削除
rcd < 0 ○
それ以外 ○
system-error
削除処理成功
元の進捗フラグ=1 and 変更フラグ=d ○
検査計画表削除
rcd < 0 ○
それ以外 ○
system-error
分析メモ削除
rcd < 0 ○
それ以外 ○
system-error
元の進捗フラグ=0 and 変更フラグ=1 ○
検査計画表削除
rcd < 0 ○
rcd == 0 ○
rcd > 0 ○
system-error
該当データなし
planout.kikaout
⇨ memoin
cnt = 1, 15 ＊
項目マスタ読込
rcd < 0 ○
rcd == 0 ○
rcd > 0 ○
system-error
該当データなし
kmkcd ⇨memoin.kmkcd kmkn⇨memoin.kmknm tani⇨memoin.tani
変更後進捗フラグ == 1 ○
変更後進捗フラグ == 2 ○
検査計画表更新1
検査計画表更新4
rcd < 0 ○
それ以外 ○
system-error
rcd < 0 ○
それ以外 ○
system-error
分析メモ追加1
rcd < 0 ○
それ以外 ○
処理が完了しました

図4. 検査計画のJSD

４．１．２　ＹＰＳの導入

上記プログラム構造作図法に基づいて作図した設計図から，関数を作成しプログラムを製作していく段階で直接 C言語を使用するとやはり生産性が落ちることから，関数管理，プログラム編集等に使用するCASEツールの調査・検討をおこなった．

その結果，「MS-DOSのYPS 」は，関数を日本語名称で登録，管理ができること，さらに当時としては画期的な「MS-DOSでマルチ・ウィンドウを採用していた事」などから，大きな生産性と信頼性を期待できると判断して導入した．その効果は予想以上であった．

４．１．３　サーバー連携システムの考案

図３に示すように、三台のサーバは，個々にホストコンピュータにデータ転送を行うことにしたので，サーバー間データ転送は無駄になるので行わない．しかし，これでは「クライアント」が「自分の親でないサーバ」のデータを使用する特殊検査表出力等のクライアント業務はできないことになる．そこで，「クライアント」からデータ検索するプログラムが起動された「親サーバ」は「自分以外のデータ」の場合，「検索マクロ」を発行して，「データを持っているサーバ」の「検索プログラム」に起動をかけて，データ収集をして貰い、「親サーバ」で加工処理を行う方式を考案した．図５に示す．

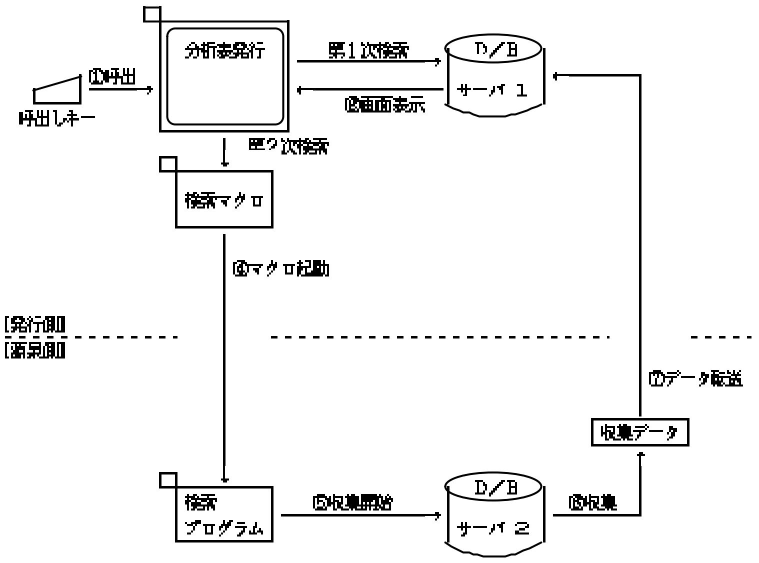

図５. サーバー連携システム

４．２システム運用の問題

４．２．１　処理レスポンスの問題

サーバの能力が小さいこともあっが，ミドルウエアは当時のパソコンには重過ぎた．これまで，F9450 系パソコンのBASIC の画面フィールド単位の敏速なレスポンスに慣れていた利用者にとっては，悠長な画面単位の処理は苦痛であった．さらにプリンタの印刷速度のおそさは，業務の遅滞の原因になるほどであった．従って，データスクリーニングの不要なアウトプットの内，バッチ的なデータ処理やデータ検索の一部については直接，INFORMIXから出力する等行ってカバーするなどの対処も行った．

サーバがエースステーション350　ΣからDS/90 に代わり，パソコンがFMR200シリーズに代わってからは，従来並の業務スピードが確保できるようになった.

４．２．２　プログラム維持の工夫

パソコンの「ＹＰＳ」でソースプログラムを生成した後，これをサーバに転送してＣコンパイラーを通して初めて実行ベースが出来上がることから，プログラム修正などの作業性が悪く，平成７年以降は「Windows 3.1 」を使用するなどテクニックに種々工夫をした.

５．新システムの効果

平成５年より順次稼働を開始し，計画通りの効果をあげている.

これについて，一部抜粋して述べる.

５．１　全社統合製品検査値管理システムの実現

各事業所，外注業者などから検査値管理システム導入の要請を受けると，短時日で，全社統合のＩＳＯ対応製品試験成績表発行システム及び，その事業所としての検査データ解析と管理のシステムを提供することが出来るようになった．これは，正に，パッケージ化＋追加型カスタマイズ方式が成功した結果である.

５．２　製造～検査～物流業務の連携合理化

従来，製造と物流と検査は業務担当者の間に，横の連携が乏しく別々の動きをしていたの対して，検査値管理システムと同時期に，別途，再構築した製造管理システム，平成７年からは，新販売物流システムとの連携による合理化も実現した．その概念図を図６に示す.

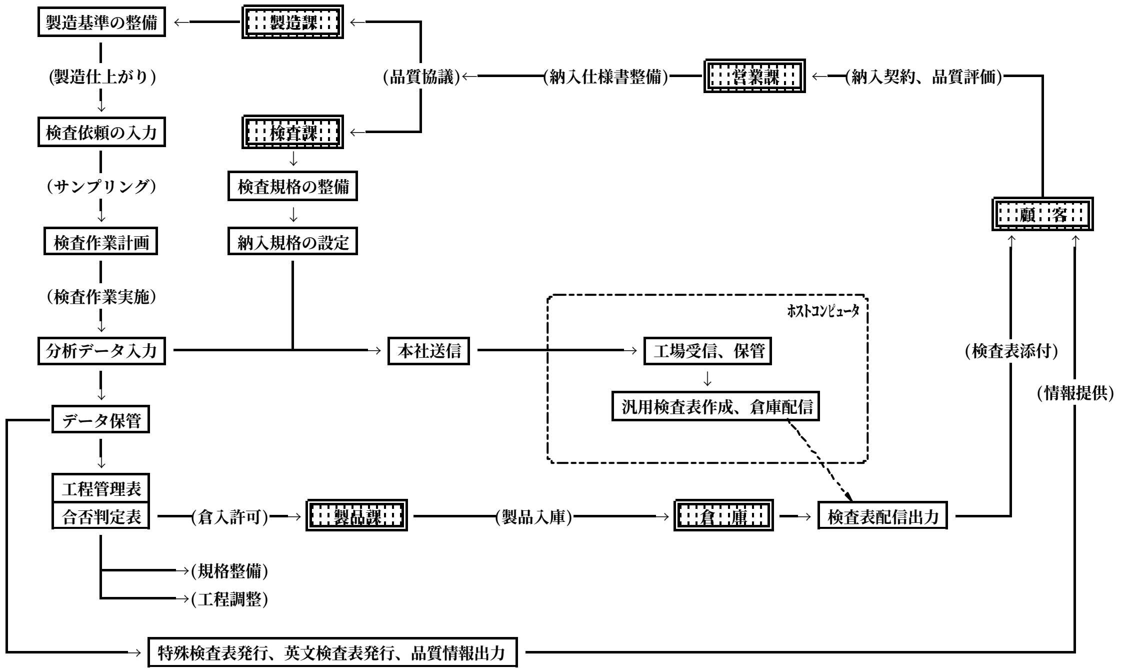

図６．検査値管理のチェーン図

５．３　検査業務自体の合理化

最初の分析値管理システムはデータ入力チェック処理から始まったもので，検査業務のスケジュール化，分析ノートの出力など，検査業務全体に展開できたことで，従来にない

合理化を実現した．その新旧対比図を図７に示す．

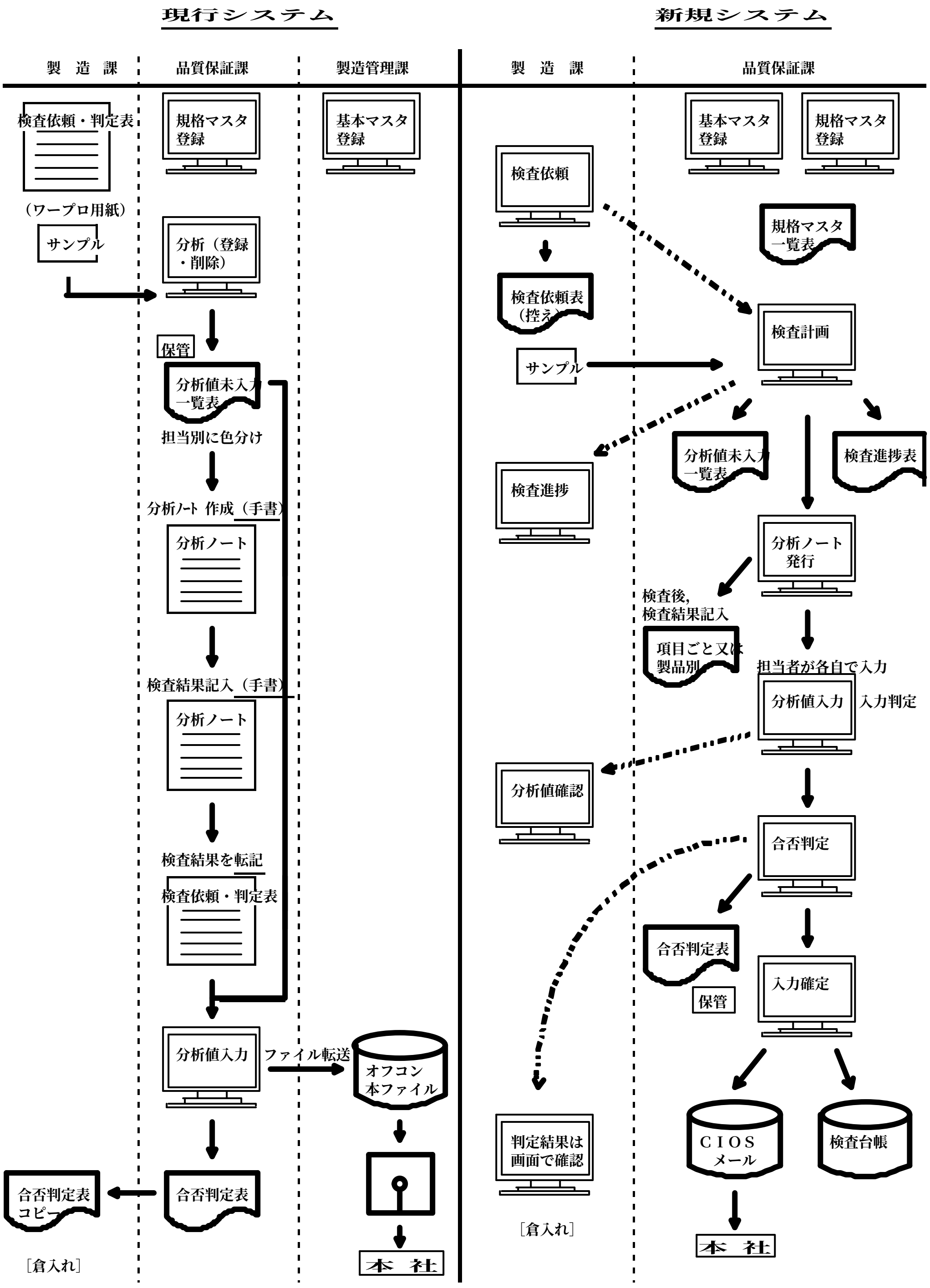

図７．三重工場での新旧対比図

６．まとめ

各事業所の製品検査値管理システムは順調に運転されており，現在も，クライアントの増設，システムの充実を図っている．

⑴パッケージ＋追加カスタマイズ方式

従来は全体をカバーするシステムを構築しておいて，エンドユーザが使用する際に，必要なメニューだけを使用する言わば「減点方式」であった．これに対して，今回方式は「加点方式」であることでエンドユーザに喜ばれており，普及も速く，開発担当ＳＥにとっても仕事がやり易く非常に好評である．

このようなことから，現在，弊社のシステム開発方針の一つに加えている．

⑵考案したプログラム構造作図法

「本構造作図法」は当初はプログラムを製作する前段階整理として考案した物である．

しかし，現在はプログラム言語に関係なくシステム設計にも適用するなど応用範囲は広がってきている．これは，本手法が対象全体を見渡しながら，ドリルダウン型で構造を明らかにしていく事（分析）と同時に，組み立てまでいける事（合成）にあるからと考える．

⑶データ提供の新しいシステムの開発

パソコン１台／人時代に合わせて，検査値管理システムに関してもこれまで進めてきたお仕着せ型システムから非定型情報提供システムへの拡充も推進していく考えである．