

データベース管理

——鈴木自動車工業株式会社における事例——

Data Base Administration of Suzuki Motor Co. Ltd.

鈴木自動車工業株式会社では、5箇年にわたる長期計画に基づいて総合部品表データベースを核とする生産情報システムの構築を推進することになった。

現在、総合部品表データベースへの移行がほぼ完了し、生産情報システムが本格的に稼動しようとしている。

本論文では、データベースシステムの構築に当たり、アプリケーションプログラム開発の生産性向上及びデータベース管理を目的として開発したSAMICSの概要について述べる。

SAMICSはアプリケーションプログラムの開発、ADM環境下の各種定義情報の一元管理及び運用支援・管理の一元化などの点で有効であり、いわゆるDD/Dシステムの一つの方向を示しているものと思われる。

谷村洋光* Hiromitsu Tanimura

伊藤健彦* Takehiko Itô

1 緒言

データベースシステムの建設はむろんのこと、その運用展開はより以上の困難さを伴うことが一般的に指摘されている。これを支える概念としてシステム管理者(あるいはデータ管理者)機能が一般化され、その具体的ツールとしてDD/D(Data Dictionary/Directry)が重要視されてきている。

このような状況の中で、ADM(Adaptable Data Manager)によるデータベースシステムの建設に当たりDD/Dシステムを必要不可欠とし、SAMICS(鈴木自動車工業株式会社でのADMアプリケーションプログラム及び定義情報の維持管理システムの意)を開発した。

本論文では、SAMICSの機能及びアプリケーションシステム開発・運用との関係について述べる。

2 システム管理者とSAMICS

2.1 データベースシステムにおけるシステム管理者の役割

一般的にデータベースシステムに関するシステム管理者の役割としては、大きく以下の3点が挙げられる。

- (1) データベースの定義と編成
- (2) データベースの保全
- (3) データベースについてのドキュメンテーション

鈴木自動車工業株式会社では、このうちデータベースの定義と編成及びデータベースの保全に特に着目し、更にAP(アプリケーションプログラム)開発の生産性向上を加味し、システム管理者の役割としている。

しかし、これらの役割をすべて人手で行なうことは、管理対象が非常に広範囲であることから不可能である。また、単に「しつけ」としてではなく「仕掛け」として管理機能を働かせるためには、システム管理者に対する具体的な支援手段としてのツールが必要である。このことから、DD/Dシステムの開発が必要となった。

2.2 SAMICSの目的

システム管理者の支援ツールとしてSAMICSは次の目的のもとに開発された。

(1) AP開発の生産性向上

- (a) フォームシートに従った仕様書入力による、ディレクトリ及びプログラムエリア用ライブラリの自動生成

ADM環境下で必要となるDBD(データベース定義)、PAS(プログラムアクセス仕様)、MFB(メッセージフォーマットブロック)などのディレクトリは定義言語を知らなくても、簡便な仕様書記述により得ることができる。

また、プログラムで使用する各エリア用のコピーライブラリを自動生成する。

- (b) プログラム手続簡略化のためのインタフェースの確立
データベースのアクセス、メッセージの送受信時などの各プログラムの共通的处理手続を、インタフェースモジュールを介することにより吸収する。

(2) ディクショナリによる各種定義情報の一元管理

仕様書をデータベース化するとともに、仕様書をもとにした関連情報のデータベース化により、ディレクトリ及びクロスリファレンスの情報を提供する。

- (3) 標準化の徹底のため、仕掛け(プログラム)による規則の自動チェック

フォームシートへの記入及び記入事項のプログラムチェックにより、約束事項で守られにくい標準の徹底を図る。

(4) 運用支援・管理の一元化

出力メッセージの再送機能、オンライン処理からバッチ処理への引継データ蓄積のためのディスクログ機能、ビデオ端末標準操作機能、データベースの保全機能など運用支援・管理を一元化する。

3 SAMICSの概要

3.1 SAMICSの構成

SAMICSは下記に示すサブシステムから構成されている。

(1) SPCS(Specification System)

仕様書処理サブシステムであり、仕様書データベースの維持、ディレクトリ及びコピーライブラリの作成などの処理を

* 日立製作所中部支店

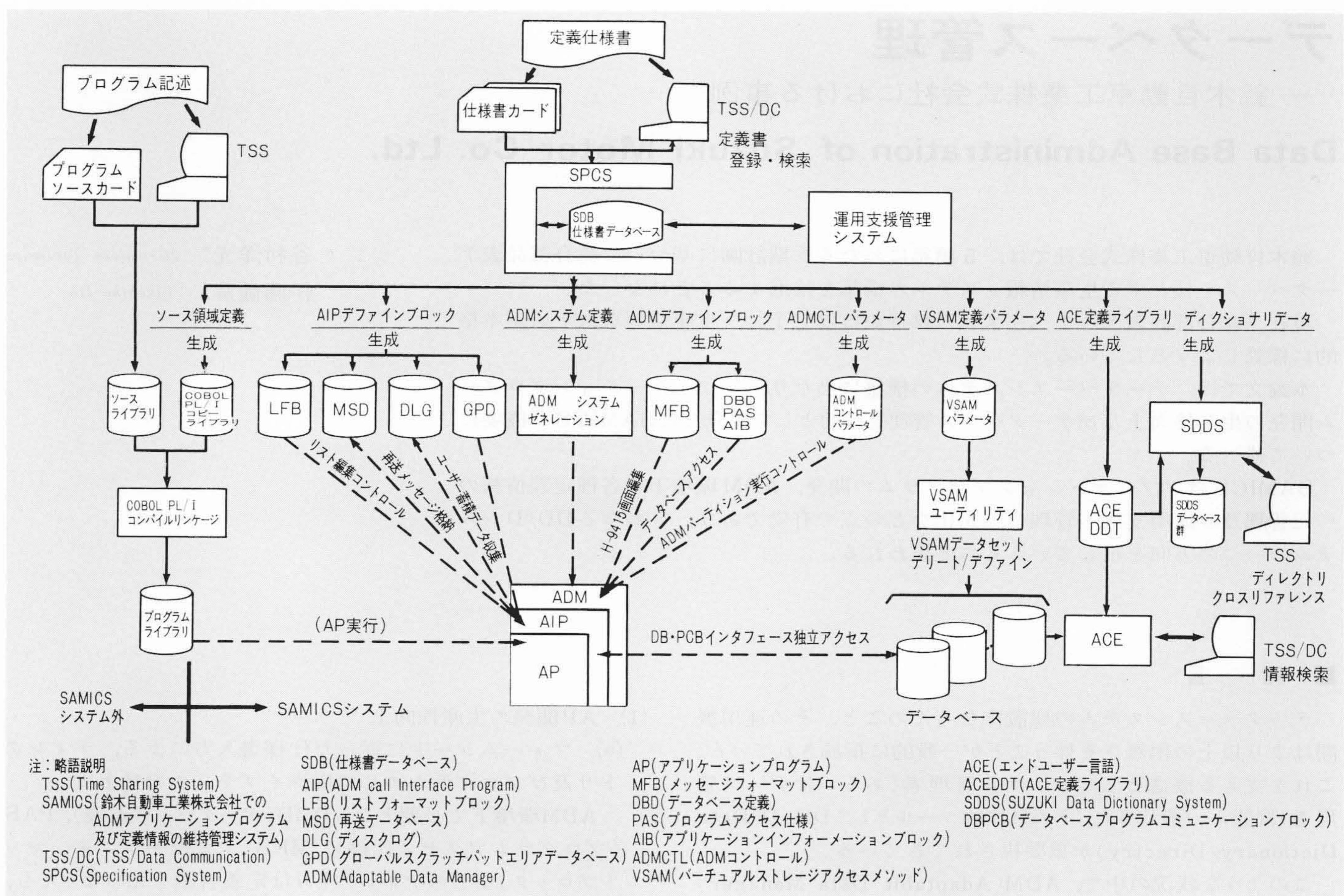


図1 SAMICSの全体関連図 定義仕様書は、仕様書データベースに蓄積維持され、各ディレクトリ出力に用いられる。

行なう。

(2) AIP(ADM call Interface Program)

データベースのアクセス、メッセージ送受信などのADMに対するAPからの要求の仲立ち処理を行なう。

(3) SDDS(SUZUKI Data Dictionary System)

ディクショナリ機能を受け持つ。

(4) 運用支援・管理プログラム群

データベース保全、ディスクログ抽出、仕様書ディレクトリ出力など種々のプログラムが用意されている。

以上の全体関連図を図1に示す。

3.2 SAMICSの機能概要

(1) ディクショナリ機能

SAMICS下の各種情報をデータベース化し、ディレクトリ、クロスリファレンスの各種情報を提供する。

(a) 定義仕様書のディレクトリ一覧

(i) 定義仕様書の定義書名を表示する。

(ii) 限定(システム、バージョン、仕様書別)定義仕様書の定義書名を表示する。

(b) 定義仕様書の内容表示

指定された定義仕様書の記述内容を表示する。

(c) プログラム又はトランザクションコードから使用データベースの表示

指定されたプログラムの使用データベース名、プロセスオプションを表示する。

(d) フィールド名称一覧

指定システム内、又は指定データベース内のフィールド名称を表示する。

(e) データベースの使用先プログラム一覧

指定されたデータベースを使用しているプログラム名を表示する。

(f) セグメントの使用先データベース一覧

指定されたセグメント名をもつデータベース名を表示する。

(g) セグメントの使用先PAS一覧

指定されたセグメントをセンシティブとするPAS名を表示する。

(2) ディレクトリ、ライブラリ自動生成機能

ADM, AIP, SDDS, ACE(エンドユーザー言語), VSAM(バーチャルストレージアクセスメソッド)で使用するディレクトリ、ライブラリ類及びプログラム用のソース領域定義のライブラリを自動生成する。

(a) ADM用に生成するもの

(i) DBD ……データベース定義

(ii) PAS ……プログラムアクセス仕様

(iii) AIB ……アプリケーションインフォメーションブロック

(iv) MFB ……メッセージフォーマットブロック

(v) ADMBMP ……BMP実行コントロールパラメータ

(vi) ADMBCH ……バッチ実行コントロールパラメータ

(b) AIP用に生成するもの

(i) LFB ……リストフォーマットブロック

(ii) DLG ……ディスクログのプリフォーマッティング

(iii) MSD ……再送データベースのプリフォーマッティング

(c) SDDS用に生成するもの

ディクショナリデータベース入力情報

(d) ACE用に生成するもの

ACEDDT ……ACE用データ定義テーブル

(e) VSAM用に生成するもの

データセット定義, 削除(カタログ, アンカタログを含む。)パラメータ

(f) プログラム用のソース領域定義

(i) データベースセグメント入出力エリア

(iii) 標準SSA(セグメントサーチアークメント)エリア

(iii) 画面メッセージ入出力エリア

(iv) リストメッセージ出力エリア

(v) 一般メッセージ入出力エリア

(vi) アクションメッセージ出力エリア

(vii) GSPA(グローバルスクラッチパットエリア)入出力エリア

(viii) SPA入出力エリア

(ix) ディスクログ出力エリア

(3) データベース保全機能

ADMシステム障害時, ディスクハードウェア障害時, AP障害時など, データベースの回復が必要となる機会が少なくない。

このためADMでは, 緊急リスタート時のデータベース回復機能を標準ユーティリティとして, データベース回復(ADM REC, ADMBAC), データベースのバックアップイメージコピーの取得(ADMIMG)が用意されている。

データベースの回復を行なうには, 入力の決定及びJCL(ジョブ制御言語)の作成を正しく行なわねばならない。しかし, データベースの回復は緊急を要し, 人間の判断・作業が多いと誤る危険性が高い。このため, 最小限の人間の介入によりイメージコピー取得, データベースの回復が行なえるように, 一貫したデータベース保全の運用支援機能を備えている。

(a) バックアップイメージコピー取得磁気テープの世代管理

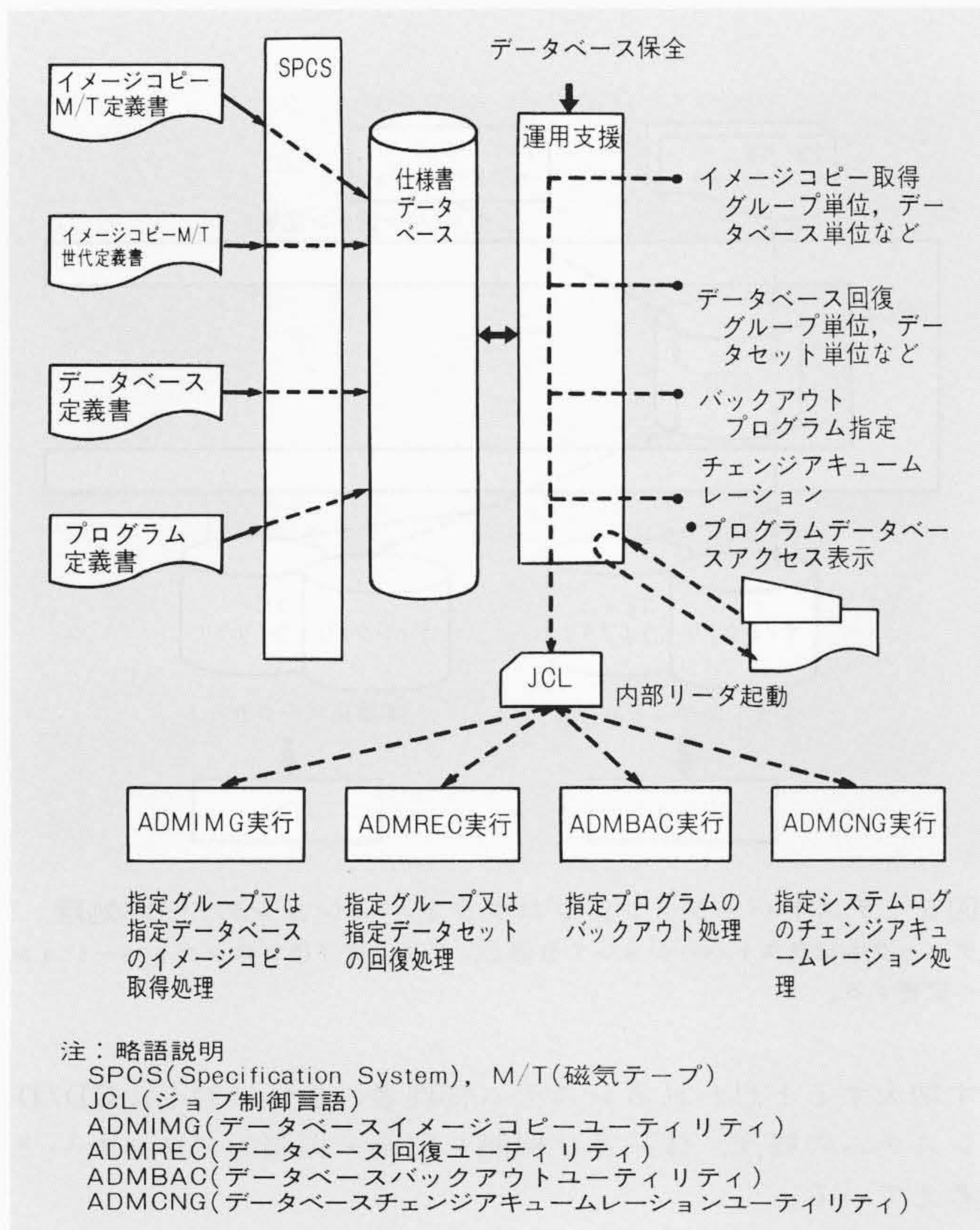


図2 データベース保全概要図 仕様書データベースの情報により, 運用支援プログラムによってデータベース保全処理用のJCL(ジョブ制御言語)を作成, 実行する。

データベースのバックアップイメージコピーの取得磁気テープを世代管理するものである。

(b) JCL自動作成

ADMREC, ADMBAC, ADMIMG, ADMCNG実行時, パラメータで最小限の情報を指定することにより, 仕様書データベースにより整合性を保ったJCLを自動作成し内部リーダに出力する。

以上のデータベース保全の関連を図2に示す。

4 アプリケーション開発とSAMICS

アプリケーションシステム, プログラム開発のステップとSAMICSの関係について説明する。

4.1 システム設計段階

(1) ADMシステム定義への組込み

データベース, プログラム(PAS), GSPAなどは, ADMゼネレーションにより, ADMシステム定義されていなければならない(オンライン対象のもの)。ADMゼネレーションは, マシン運用上随时できない。SAMICSには定義仕様書入力によるADMゼネレーション処理も考慮されているが, 運用上の制限とし, ADMシステム定義への組込みはSAMICS外処理とする。

このため, データベース, プログラム(PAS), GSPAは, オンラインデバック開始3週間程度前にADM管理者に申請し, ADMシステム定義を行なう必要がある。

(2) 共用定義書の作成

プログラムで共用される定義仕様書を作成する。

- (a) データベース定義仕様書
- (b) データベースアクセス定義書
- (c) ディスクログ定義書
- (d) アクションメッセージ定義書
- (e) GSPA定義書

4.2 プログラム設計・作成段階

(1) プログラム対応で用いる定義仕様書を作成する。

- (a) プログラム(PAS)定義書
- (b) 画面定義書
- (c) 入力メッセージ定義書
- (d) 出力メッセージ定義書
- (e) リスト定義書
- (f) アクションメッセージ定義書

4.3 プログラムデバック段階

(1) データベースのバックアップグループの登録

- (a) データベース定義仕様書の修正

以上, 各仕様書の開発経緯との関連を図3に示す。

4.4 デバック支援

(1) プロダクトバージョンとテストバージョン

定義仕様書入力により, ディレクトリ, コピーライブラリの自動生成を行なうSPCS処理には, プロダクトバージョン, テストバージョンの二つのバージョンがある。バージョン指定は定義仕様書入力時(修正入力を含む。)に行なう。

プロダクトバージョンを指定されたものは, SPCS処理時, 本番用のデータセットにディレクトリ及びコピーライブラリを出力する。このため, 日中のオンライン稼働時には, 本番用のデータセットを更新するプロダクトバージョンのSPCS処理を行なうことはできない。

プログラムデバック段階に必要とするディレクトリ, コピーライブラリの作成(修正)が, オンライン終了後の夜間1日1回では, プログラム開発に多くの日数を要してしまう。そ

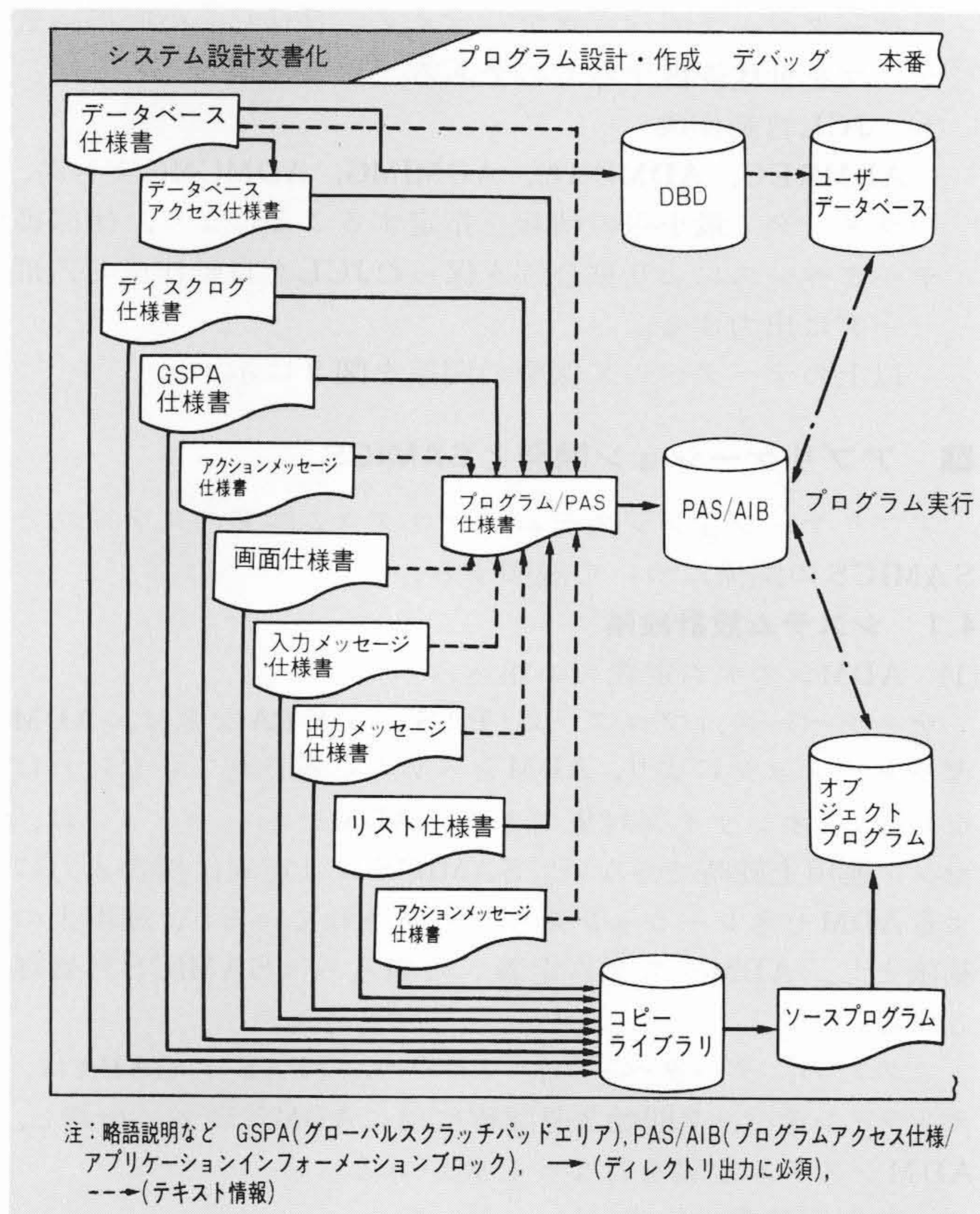


図3 システム開発ステップとSAMICS定義仕様書作成関連
SAMICSの各定義仕様書は、システム開発のステップにより順次登録される。

ここでプログラムの開発日数を短縮するために、デバッグ支援機能としてサポートするのがテストバージョンである。テストバージョン指定の定義仕様書のSPCS処理では、本番用のデータセットとは異なるテスト用のデータセットにディレクトリ、コピーライブラリを出力する。

このテストバージョンを使うことにより、オンライン中であってもプログラムデバッグを行ない、必要ならばSPCS処理によりディレクトリ及びコピーライブラリの作成(修正)を行ない、効率の良いプログラムデバッグが可能である。図4にデバッグ処理の関連を示す。

(2) テスト、プロダクトバージョン運用

定義仕様書の初期入力時はテストバージョンとして入力し、仕様書のチェック、出力ディレクトリ、コピーライブラリのチェック、プログラムのデバッグ後テストバージョンをプロダクトバージョンに変更し、本番用のディレクトリ及びコピーライブラリを作成する。

図5にバージョン変更の関連を示す。

5 結 言

以上、データベースシステムの建設・運用に当たって鈴木自動車工業株式会社で開発したSAMICSについて述べた。

SAMICSは、(1)アプリケーションプログラム開発の生産性向上に効果がある(従来に比較して約30%程度向上できた)。(2)ADM環境下でのソフトウェアリソースを一元管理することができる。(3)データベース保全に著しく効果を挙げることができる。などの特長があり、DD/Dシステムとして既に定着し成果を挙げている。

現在、SAMICSはまだ制約があるが、DD/Dシステムとしての一つの方向を見いだしたと考えている。今後は、ますます

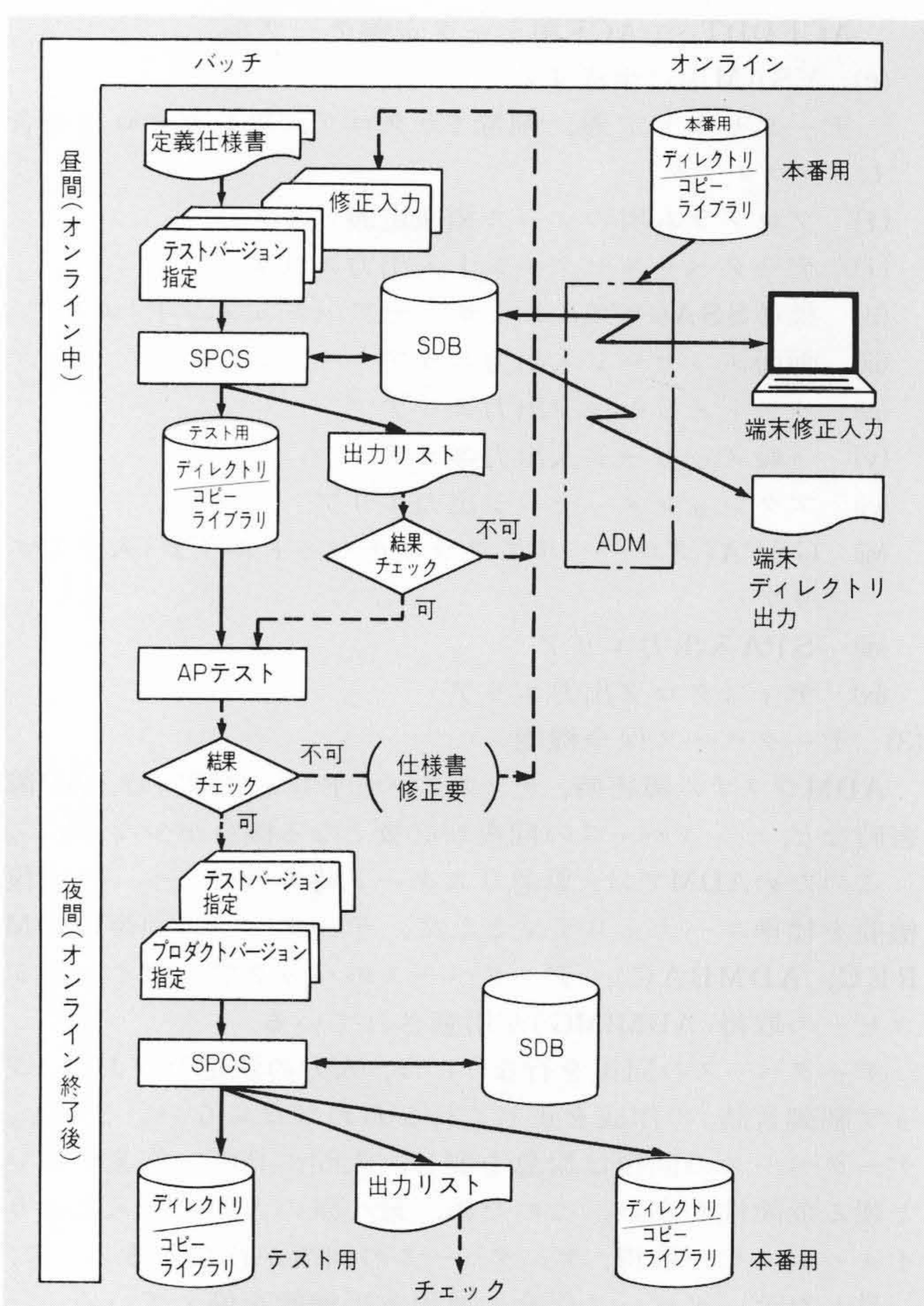


図4 デバッグ処理関連図 定義仕様書の登録からAPテスト環境によりデバッグを行ない、結果の確認後、プロダクトバージョンへ変更する。

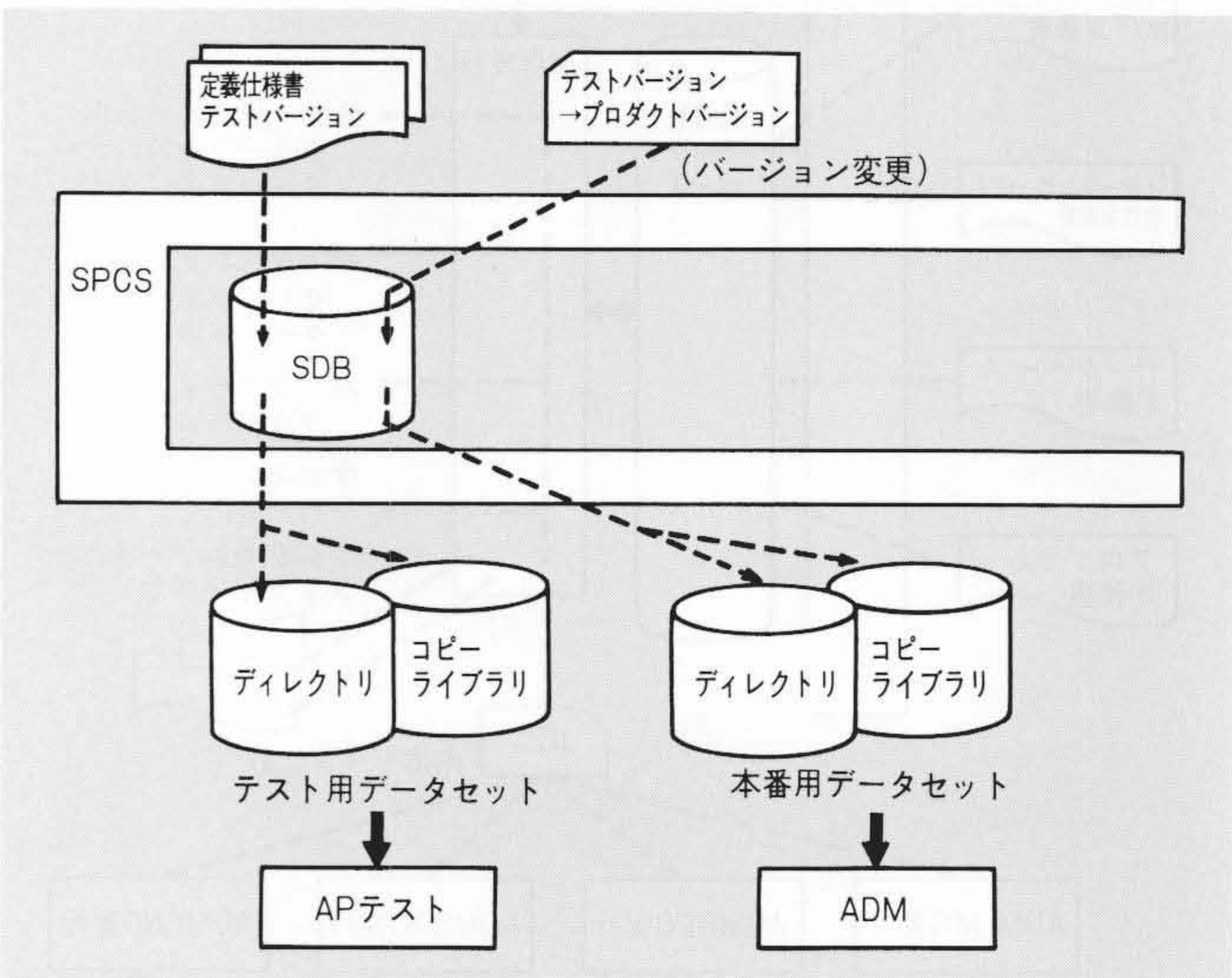


図5 テストバージョン、プロダクトバージョンのSPCS処理
デバッグ中はテストバージョンで登録し、テスト完了後プロダクトバージョンへ変更する。

す増大すると思われるシステム管理者の責任に対し、DD/Dシステムの概念、体系及び機能を整備・拡張してゆきたいと考えている。

本論文の執筆に当たり、鈴木自動車工業株式会社の御好意によりシステムの公開及び執筆への御協力をいただいたことについて深く御礼申し上げます。