

リアルタイム処理向きファイル管理システム “FMS-R”

File Management System for Realtime Data Processing “FMS-R”

大脇隆志* *Takashi Ôwaki*
 広瀬 正** *Tadashi Hirose*
 林 利弘* *Toshihiro Hayashi*

計算機制御システムでは、制御対象プロセスへの追従性、制御の連続性が必要不可欠の要件であり、システムの高応答性と高信頼化が重要である。更に、制御用計算機の処理能力向上に伴い、「制御」と「情報処理」の混在など、データ処理は複雑化、多様化の傾向にあり、アプリケーションソフトウェアの生産性、保守性向上は、以前にもまして重要な課題となっている。

日立製作所では、このような状況に対処するため、高性能・高信頼性を特徴とするリアルタイム処理向きファイル管理システムFMS-Rを、HIDIC V90シリーズの基本ソフトウェアの一環として開発した。FMS-Rは、従来の補助記憶ファイルに加えて主記憶ファイルをも統一的にサポートした「バーチャルなファイルインタフェース」と、アプリケーションプログラムに独立な「オンライン性能調整機能」とによりソフトウェアの生産性、保守性及び信頼性の向上を図っている。

1 緒 言

最近のLSI技術、コンピュータアーキテクチャの進歩には目覚ましいものがあり、年々、計算機の記憶容量は指数関数的に増大している。この傾向は、制御用計算機やマイクロコンピュータでも例外ではなく、主記憶は一昔前のkバイトのオーダーから、Mバイト、Gバイトのオーダーへと移り変わってきている。

このような状況の中で、計算機に対する処理要求は多様化しており、大容量主記憶をベースとし、処理装置の能力を十分に発揮する高性能ミニコンピュータ及びマイクロコンピュータの実現要求が高まっていた。

しかし、これを実現するに当たっては、従来のテーブル操作的なソフトウェアの作り方では大容量主記憶を扱いきれないこと、一方、従来のファイル管理システムは補助記憶しか扱えないことなど、主記憶をベースとするコンピュータアーキテクチャテクノロジーに対応すべきソフトウェアのサポートが十分ではなかった。

このようなことから、補助記憶に加えて主記憶をもファイルとして統一的に扱うことを可能とし、ミニコンピュータ、マイクロコンピュータでの高性能リアルタイム処理の実現と、アプリケーションソフトウェアの開発を支援する、標準的なソフトウェアのサポートが強く望まれていた。これを実現するため、HIDIC V90シリーズの基本制御ソフトウェアの柱の一つとして開発したものが、FMS-R(File Management System/Realtime Version)である。

本稿では、FMS-R開発の基本思想、開発方針、構成、機能及び適用結果について述べる。

2 FMS-R開発の基本思想

高性能リアルタイム処理システム実現に当たっては、

- (1) 大容量主記憶を利用したリアルタイム高応答性能の達成
- (2) 電源ダウン時の主記憶データ消失の防止による、処理及びデータの連続性保証

が重要課題となってくる。

また、これらを実現するアプリケーションソフトウェア開発のしやすさといった面からは、

- (3) 主記憶～補助記憶の間の境界を排除した、一様なデータ空間の扱いを可能とすること。
 - (4) システムの拡張や変更に対して柔軟に対処できること。
- が必要不可欠の要件となる。

FMS-Rでは、これら高性能リアルタイム処理システムに対する前記(1)～(4)の要件に基づき、以下を開発の基本思想とした。

- (1) 主記憶活用を前提としたファイル処理高速化アーキテクチャの実現を図ること(高性能)。
- (2) システムの信頼性を向上させる、ファイル高信頼化機能を具備すること(高信頼性)。
- (3) アプリケーションプログラムからはファイルの物理構造を意識させないバーチャルなファイルインタフェースを与えること(使いやすさ)。

次章ではこれらの基本思想に基づく、FMS-Rの開発方針について述べる。

3 FMS-Rの開発方針

(1) 高性能化

- (a) きめの細かいファイル共用管理によるタスク並行処理性の向上

レコード単位のファイル共用を可能とすることにより、ファイルへの同時アクセスに伴う、複数タスク間の競合発生頻度を極小化する。これにより、タスクの並行処理性を確保し、タスク応答性の向上を図る。

- (b) プリコンパイル処理の徹底による実行時オーバーヘッドの削減

アプリケーションプログラムからファイルへのアクセス時ごとに参照されるファイルディレクトリ テーブルなどに

* 日立製作所大みか工場 ** 日立製作所システム開発研究所

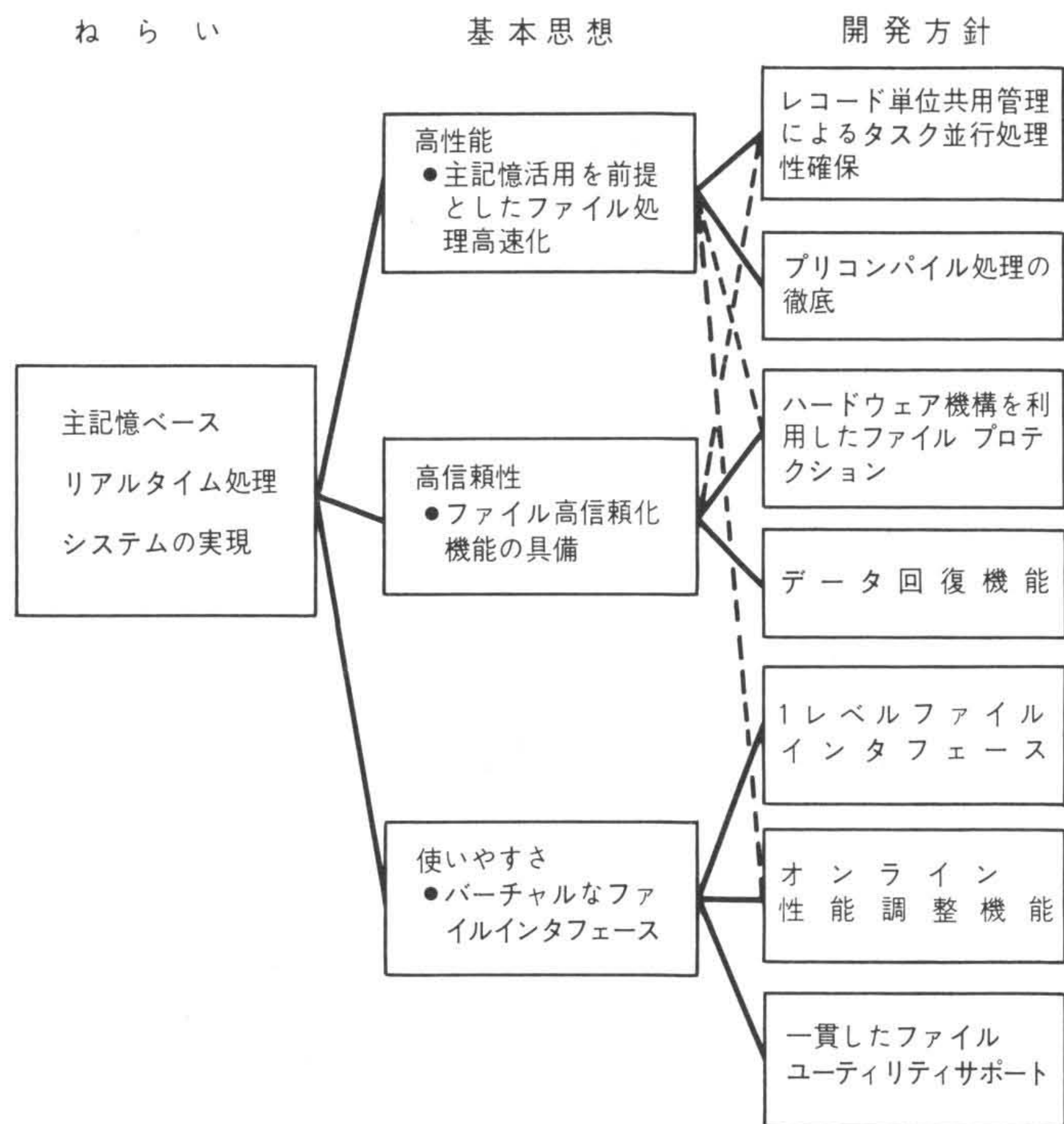


図1 FMS-R開発の基本思想と方針 FMS-Rは、主記憶をベースとした高性能リアルタイム処理システムの実現をねらいとし、「高性能」、「高信頼性」及び「使いやすさ」を基本思想の3本柱として開発した。

については、これの所在アドレスをあらかじめプログラムオブジェクト中に展開しておくことなど、プリコンパイル処理の徹底によりリアルタイム実行時のファイル処理オーバーヘッドを削減する。

(2) 高信頼化

- (a) ハードウェア機構を利用したファイル保護機能の実現
メモリプロテクション機構などのハードウェアの機能を十分に活用することにより、ファイルアクセスチェックな

どの実行時オーバーヘッドを極小化しながら、ファイル破壊を未然に防止する。

(b) データ回復機能の具備

万一、ファイル障害やデータ消失が発生した場合にも、これを高速に回復するためのファイルリカバリ機能を具備することにより、システム稼働率の向上を図る。

(3) 使いやすさの向上

(a) 1レベルファイルインタフェースの実現

従来プログラマが意識しなければならなかった、補助記憶ファイルアクセスのための「バッファ」を、プログラムから分離しシステムが統一的に管理することにより、主記憶及び補助記憶の違いを意識させないようなファイルインタフェース(「1レベルファイル」インタフェース)を実現する(図2)。これにより、プログラム作成の容易化を図る。

(b) オンライン性能調整機能の具備

プログラムインタフェースからファイルの物理構造情報などの性能要因を排除すると同時に、これら性能要因を、アプリケーションプログラムを一切変更することなく、調整可能とすることにより、システム性能調整の容易化を図る。

(c) 一貫したファイルユーティリティのサポート

ファイルの生成、テストから保守、運用管理に至るまで様々な局面に応じて必要となる、種々のファイルユーティリティ機能をきめ細かにサポートすることにより、システム開発効率、保守性・拡張性の向上を図る。

4 FMS-Rの構成

以上に述べたFMS-Rの開発方針を実現するためのシステム構成を図3に示す。FMS-Rは、次の三つのサブシステムから構成される。

(1) 実行時サブシステム

リアルタイム処理実行時に、アプリケーションプログラムとオンラインファイルの間のデータ授受を統一的に実行する部分である。レコード単位のデータ操作を行なう基本ファイルアクセスマクロとその他の拡張ファイルアクセスマクロか

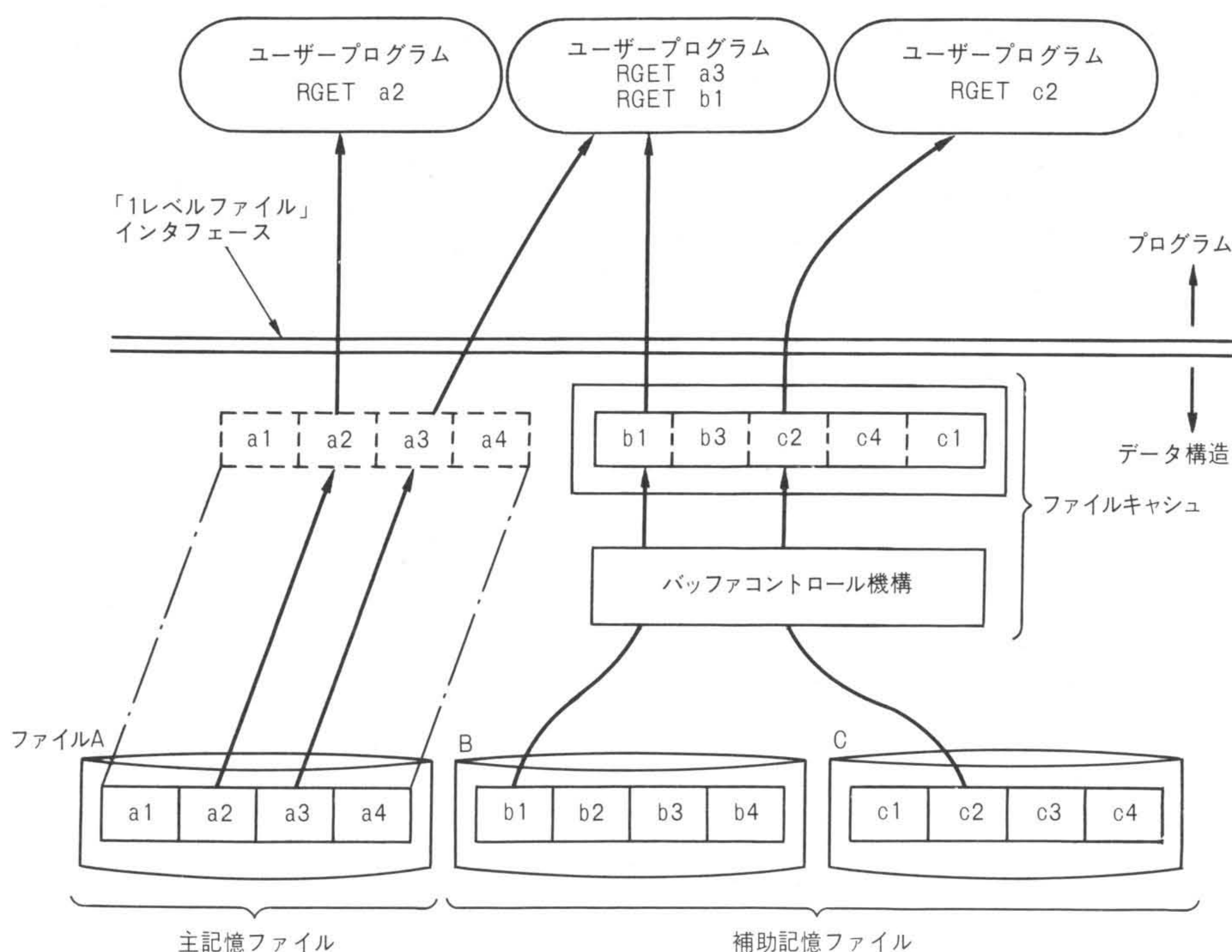


図2 「1レベルファイル」プログラミングインタフェース アプリケーションプログラムからは、主記憶ファイルと補助記憶ファイルの違いを意識させない、バーチャルなファイルインタフェースでファイルアクセス可能とする。

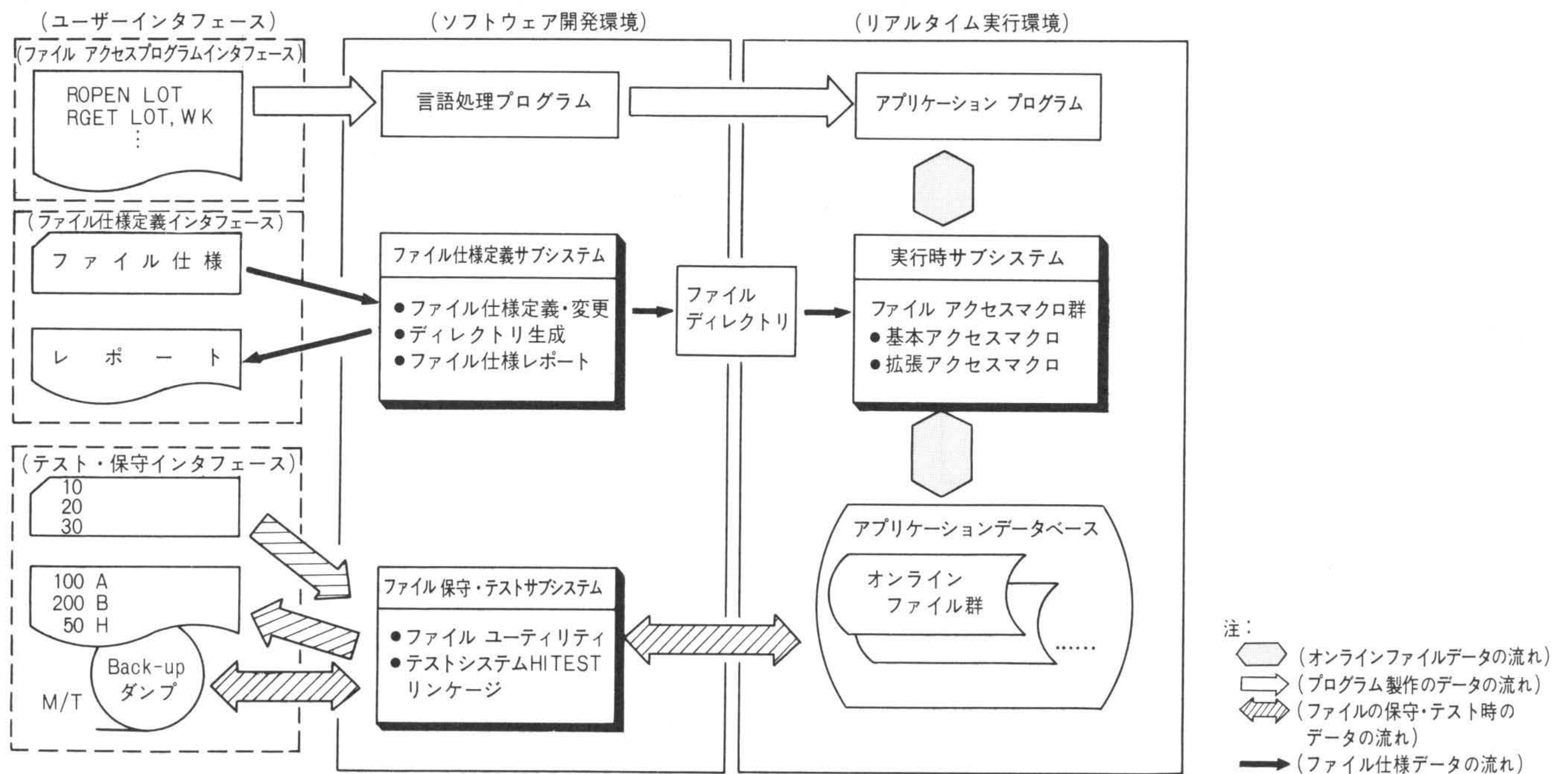


図3 FMS-Rのシステム構成 FMS-Rは「実行時サブシステム」、「ファイル仕様定義サブシステム」及び「ファイル保守・テストサブシステム」の三つのサブシステムから構成される。

表1 FMS-Rファイルアクセスマクロ一覧 FMS-Rのアプリケーションプログラムとのインタフェースは、レコード単位の操作を行なう基本マクロ群とフィールド操作、ソートなどの拡張マクロ群から成る。

分 類	マクロ名称	機 能	分 類	マクロ名称	機 能		
基本マクロ	宣 言	RLINK	ファイル アクセスのためのリクエスト ブロックを作成する。	ドレ 操 作 ↓	RERASE	レコードをファイルから削除する。	
		ROPEN			RPOINT	処理開始位置を指定する。	
		RCLOSE					
	レ コ ー ド 操 作	RREFER	指定されたレコードを読み出す。	操 作 ↓ フ ィ ー ド	RUPDTF	レコード内のフィールドを更新する。	
		RADD			RTRNSF	レコード内のフィールドを参照する。	
		RGET			ソ ー ト	RSELECT	ファイルからソート対象レコードを条件付きで選択する。
		RPUT				RSORT	ファイル内レコードを指定されたキーに従いソートする。
		RUPDAT				RORDER	ファイルをインデックス ファイルに従い並べ替える。
			その他	RGTATRなど	ファイル属性取り出しなど。		

ら成る。

FMS-Rのファイルアクセスマクロの一覧を表1に示す。

(2) ファイル仕様定義サブシステム

ファイルの物理構造やアクセス性能をアプリケーションプログラムとは独立に定義・変更・調整を可能とするサブシステムである。ここで定義されたファイル仕様情報については、リアルタイム処理実行時にファイルアクセスマクロが高速に参照可能なように、ファイルディレクトリとして主記憶上に生成しておくことにより、リアルタイム処理実行時のオーバーヘッドを削減できるようにしている。

(3) ファイル保守・テストサブシステム

オンラインファイルの保守やオンラインファイルをアクセスするアプリケーションプログラムのテストを行なうためのサブシステムである。各種のファイルユーティリティと、制御用ソフトウェアのテストを支援するテストシステム“HITEST”(Hitachi Integrated Test System)とのリンケージ部から成る。

5 FMS-Rの機能

5.1 実行時機能

アプリケーションプログラムとオンラインファイルとの間のデータの授受を、効率良く、かつ信頼度高く実行すること

を支援する機能である。

(1) 多様なアクセスパスの提供

リアルタイムデータ処理システムでは、処理対象データに関連して、マンマシン処理、プロセス制御、実績集計処理など様々なアプリケーション処理に対応したデータアクセス法や異なるアクセス性能が必要となる場合が多い。FMS-Rでは、こういった多様なデータアクセス要求を満たすために、表2に示すような5種類のファイル編成をサポートしている。

これら五つのファイル編成を組み合わせることで、より、多様なアクセスパスが実現できるので、ユーザーは、アプリケーションの特徴に適合した効率の良いデータベースを構築できるようになる。

(2) 主記憶ファイルの実現

FMS-Rでは、

(a) 厳しい応答性能を要求されるデータファイル

(b) 頻繁にアクセスされるインデックスファイル

などを「主記憶ファイル」とすることにより、応答性、処理性の向上を図れるようにしている。これと同時に、主記憶ファイル上のデータは、

(a) アクセスチェックによる不当なデータ書込みの排除

(b) データ消失時にもファイルリカバリによりデータを復

表2 FMS-Rのファイル編成一覧 FMS-Rは、基本、相対順、ハッシュ、サイクリック及び索引順の5種類のファイル編成をサポートすることにより、リアルタイムアプリケーションの種々のデータ処理を統一的に扱えるようにしている。

項目	編成	基 本	相 対 順	ハ ッ シ ュ	サイクリック	索 引 順
論理構造		RRN 1 2 3 4 ⋮	RRN 1 4 5 6 ⋮	(RRN) KEY1 → 1 KEY2 → 2 KEY3 → 6 ⋮	(RRN) 1 2 3 4 5 6 注：↑↓ 追加，削除	KEY 1 10 2 20 3 25 ⋮ 30 45 ⋮
特 徴		物理アドレス (RRN) による高速アクセス	レコード空実管理機構	Hashingによる高速なレコード検索	サイクリック管理機構による高速な追加削除	B ⁺ -tree*によるレコード検索
適 用		●追加削除のないファイル (例) ●設備条件ファイル ●トラッキングファイル	●ランダムな追加・削除のあるファイル (例) ●計画ファイル ●実績ファイル	●ランダムなKEYによるアクセスのあるファイル (例) ●問い合わせ用インデックスファイル	●順序(FIFOなど)に基づく追加，削除のあるファイル (例) ●トラッキングファイル (ベルトコンベヤなどの) ●通信バッファ	●KEYの内容・順序に基づく検索のあるファイル (例) ●一般情報処理
アクセス法		○	○	△	△	△
RRN直接		○	○	△	○(時間順)	△
RRN順		×	×	○	×	○
キー直接		×	×	×	×	○
キー順		×	×	×	×	○

注：○(アクセス可)，△(追加・削除は不可)，×(アクセス不可)
B⁺-tree:データをその値の順にn分木状に配置した索引構造をいう。キー順，キー直接両アクセス法による高速な索引を可能とする。

元可能

など、高信頼化が図られるようになる。

(3) ファイルキャッシュ機能

従来、アプリケーションプログラムは、補助記憶ファイルへのアクセス時、補助記憶とのデータの入出力のための緩衝域(バッファ)を自プログラム内にもち、これを意識した記述となっていた。このため、

(a) バッファサイズなどの性能要因を各プログラマが考慮

しながらプログラミングせざるを得ず、煩雑かつ誤りやすい。

(b) 性能調整を行なう場合、個々のプログラムの修正が発生する。

という問題があった。FMS-Rでは、こういった問題を解決するために、バッファをアプリケーションプログラムとは独立にファイル側にもたせ、これをシステムで統一して管理する「ファイルキャッシュ機構」を具備することにより、アプリケーションプログラムにはバッファを意識させないバーチャ

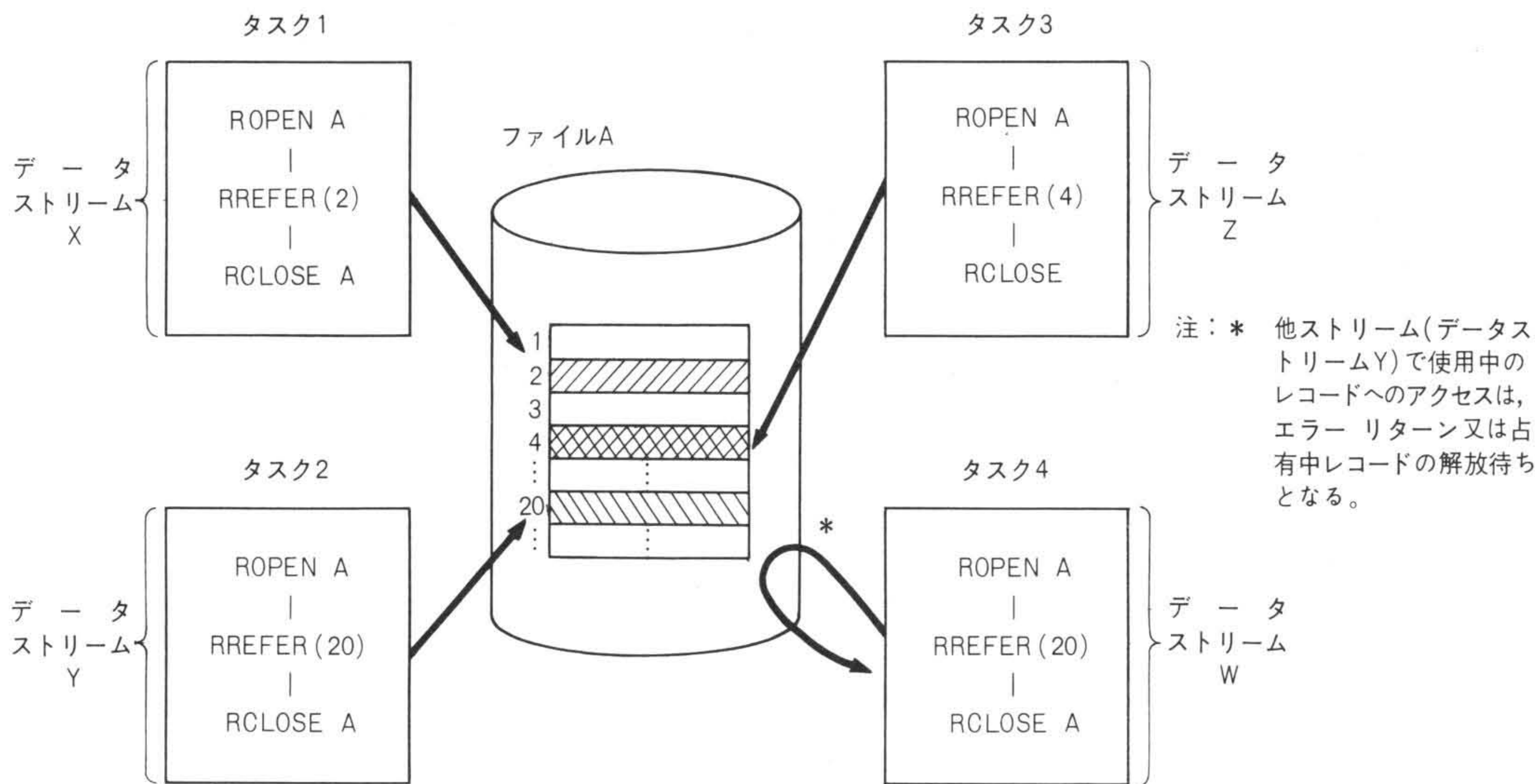


図4 マルチデータストリーム機能 FMS-Rでは、1個のファイルを複数のタスクが同時に使用できるようにし、かつレコード単位の排他制御機能により、データの競合によるファイルの破壊を防止している。

ルなファイルインタフェースを実現している。これにより、
(a) バッファなどの煩雑な手続きからプログラマが解放される。
(b) 個々のプログラムを変更することなく性能調整を行なえる。

といった効果が得られるようになる。

(4) マルチデータストリーム機能

ファイルキャッシュ機構のもう一つの効果は、バッファがプログラムとは独立にファイル側にあるため、バッファ上のデータを複数のタスクが共用できることである。これにより、従来、データの排他制御は、バッファと補助記憶との入出力の単位(ブロック)で行なわざるを得なかったものが、ブロック単位を意識せずにレコード単位で行なえるようになる。FMS-Rでは、これを利用して、複数のタスクが同時に同一のオンラインファイルを使用可能とし、かつタスクの競合に伴うデータ破壊を、レコード単位の共用管理機能により防止する「マルチデータストリーム機能」を実現している(図4)。

このマルチデータストリーム機能によれば、タスク並行処理性の確保とタスク応答性の向上が図られるとともに、レコード単位のきめの細かいデータ共用管理により、データの信頼性向上が図られるようになる。

(5) ファイルリカバリ機能

ファイル障害の要因としては、
(a) CPU(中央処理装置)停止、電源ダウンなどハードウェアの故障によるもの。
(b) プログラムエラーなどソフトウェアの故障によるもの。
(c) データの誤入力など、オペレーション上のミス(人的要因)によるもの。

などがあり、FMS-Rでは、これらのファイル障害要因に対応して、様々なファイルリカバリの形態がとれるようにしている。FMS-Rでのファイルリカバリの方法は、ファイル修復時間の長さにより大きく「ホットリカバリ」、「ウォームリカバリ」及び「コールドリカバリ」の3種類に分けられる(表3)。

表3 ファイル リカバリの種類と方法 FMS-Rでは、ファイルの障害要因、修復時間の大きさに応じて、ホットリカバリ、ウォームリカバリ、コールドリカバリといったアプリケーションに適したリカバリの形態をとれるようにしている。

項目 種類	対象となる故障(程度)	修復時間	リカバリの方法
ホ ッ ト リカバリ	主としてハードウェア(デバイス)局所的エラー、偶発的エラーによるファイルの異常 (小~大)	小	ファイルの二重化・リトライなど。ユーザープログラムにはファイルの異常を意識させずに回復
ウォーム リカバリ	プログラムエラーによるタスクの異常終了やオペレーションミスなどによるファイル内データの首尾一貫性異常 (小)	中	ファイルの異常な状態から、ファイルに対する更新履歴情報により以前の正常な状態へ復帰
コールド リカバリ	デバイスダウン、メモリ電源断などによるファイル全体に及ぶ異常 (大)	大	ファイルのバックアップコピーとファイル更新履歴情報をもとに、任意の時点のファイル状態へ復帰

ユーザーは、各々のアプリケーションシステムの運用方法やデータの性質に応じて、これらファイルリカバリの方法を組み合わせて用いることにより、高信頼度のシステムを構築できるようになる。

5.2 開発・保守支援機能

システム開発、保守、拡張といった、ライフサイクル全般にわたるアプリケーションデータベースの効率的な運用・管理を支援する機能である。

(1) ファイル仕様定義とオンライン性能調整機能

リアルタイム処理アプリケーションシステムのファイル仕様をプログラムとは独立に定義し、また、システム性能の調整をオンラインへの影響を最小に行なう機能である。

ファイル仕様定義では、
(a) オンラインファイルの格納装置種別(主記憶及び補助記憶)、ファイルキャッシュの仕様(バッファのサイズ、面数)などの性能設計情報
(b) オンラインファイルのサイズ、レコード件数などの物理設計情報

をシステム設計者が一括して管理、定義できる。またこれらは、アプリケーションプログラムに一切手を加えることなく変更することができる。

このことにより、
(a) システム構築後の性能調整もプログラムの変更なしに容易にできるようになる。
(b) ファイルの拡張など増設、改造に対してもシステムを止めることなく柔軟に対処可能である。

など、システムの保守性、拡張性の向上が図られる。表4にファイル仕様定義項目の一覧を示す。

(2) ファイル保守・テスト支援機能

オンラインファイルの信頼性、保守性向上のためには、ファイルの生成・保守、プログラムテスト、システムの運用・管理といった各目的に対応した「ファイル保守・テスト支援機能」の充実が不可欠であり、FMS-Rでは図5に示すよう

表4 FMS-Rファイル仕様定義項目 FMS-Rファイル仕様定義機能では、これら定義情報を、アプリケーション プログラムと独立に、定義・変更・削除できる。

分 類	仕 様 定 義 項 目
物理ファイル 仕 様 定 義	● 物理ファイル構造仕様 (ファイル編成種別、レコード長、ブロック長、レコード数など) ● キー情報 (キーのタイプ、キー長、キー ポジション) ● ファイル リカバリ仕様 (リカバリ ログ取得指示、リカバリ クラス番号、リカバリ方式)
バッファ仕様 定 義	● ファイルキャッシュ仕様 (バッファブロック サイズ、バッファブロック数など)
論理ファイル 仕 様 定 義	● 物理ファイル、ファイルキャッシュとの対応付け ● 論理ファイル属性 (ファイル共用タイプ、ファイルプロテクション タイプ) ● レコード タイプ (データ形、出力フォーマット)

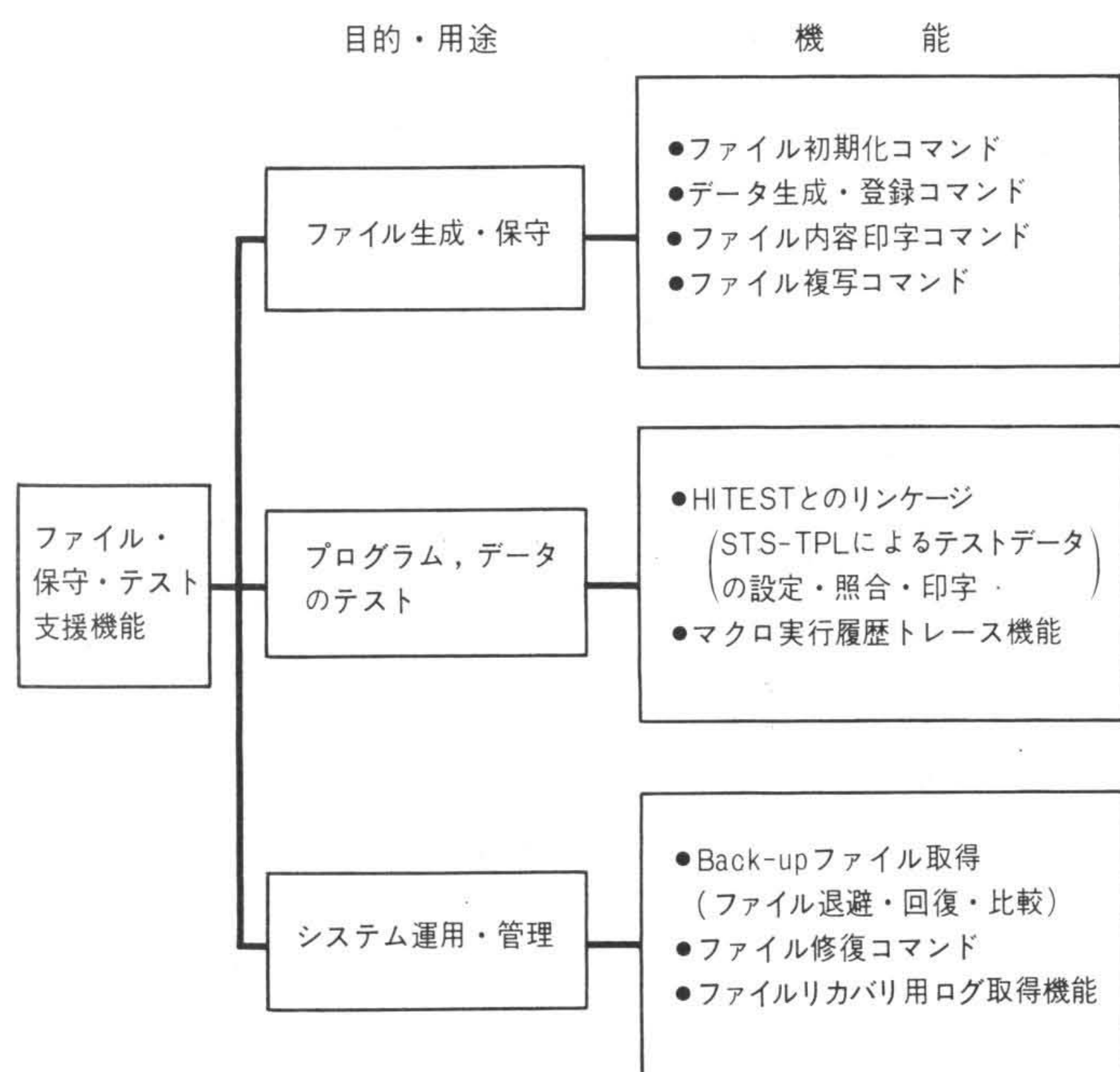


図5 FMS-Rファイル保守・テスト支援機能 FMS-Rは、ファイル保守、アプリケーション プログラムのテストのため、その目的に応じた種々のユーティリティを装備している。

な豊富な支援ユーティリティを装備している。図6は、その一つであるファイル内容印字ユーティリティのフォーマット指定編集機能による印字出力例である。本ユーティリティでは、レコードタイプに従った印字フォーマットをあらかじめ登録しておくことにより、ファイル名とフォーマット番号とを指定するだけでファイル内容を編集して、見やすく印字出力させることができ、簡単な帳票であれば、プログラミングなしで容易に作成することが可能である。

更に、FMS-RではテストシステムHITESTとのリンケージ機能により、ファイルへのテストデータの設定、照合、印

字などアプリケーションプログラムのテストを、HITESTの種々のテスト機能を利用して行なえるようにしている。これにより、テスト作業が自動化、標準化されるなど、プログラムテストを効率良く行なえるようになる。

6 FMS-Rの適用状況と結果

FMS-Rは、昭和58年4月にHIDIC V90シリーズ用として完成し、以来、鉄鋼、生産管理、電力、火力、上下水道、FA(ファクトリーオートメーション)などリアルタイム処理アプリケーションの広範な分野にわたり適用されてきた。

適用結果としては、

- (1) 主記憶ファイルへの高速なアクセスやファイルキャッシュ機構などにより、従来、標準的なファイル管理システムが適用不可と考えられていた分野に対しても、要求性能を十分に満たしており、ファイル周りの性能上のトラブル発生を大幅に減少させることができた。特に、ファイル処理時間オーバーヘッドを、従来の一般ファイル管理システムに比べて $\frac{1}{5} \sim \frac{1}{10}$ に短縮できた。
- (2) 不当なファイル アクセスやデータ破壊が未然に検出されるなど、システムの信頼性向上が図られるとともに、アプリケーションプログラムのテスト デバッグ効率が向上した、などが顕著な効果である。

7 結 言

以上、FMS-Rの開発方針、構成、機能及び適用結果について述べた。

FMS-Rはリアルタイム処理アプリケーションを対象として、高応答、高信頼化を実現しながらソフトウェアの生産性、保守性を向上させることをねらった従来にはない総合的なファイル管理システムであり、その適用を通じ、所期の目的を十分に達成していることを確認できた。

今後は、適用結果を踏まえ、更に使いやすいシステムとするための不断の努力を続けていく考えである。

(1) ファイル内容印字コマンド入力例

```
$ RPRINTF CYCF03 / LIST=LP ; /*ファイル名称 CYCF03のファイルをフォーマット
FMPRNT DAFMT=20          番号20のフォーマットに従い、編集出力する */
END
```

(2) 出力例

*** FMS-R ***		PRINT FILE UTILITY		VER 00-00			
		FILE NAME=CYCF03		FILE ORGANIZATION=CYCLIC			
NAME	コード	@ \$	@ KG	ザイコ	メーカー	レート	DATE
HIJ005	3072	103.8	110.3	42	A496	1.1	0307
GHI902	4967	102.9	926.3	3	C301	1.5	0215
FGH310	0528	982.5	824.4	10	B100	1.5	0215
EFG882	1059	286.3	921.1	19	D929	3.0	0103
DEF603	6668	186.2	110.5	18	A435	0.5	0103
CDE599	4362	100.0	392.8	29	C102	1.0	0101
BCD330	9625	304.9	232.6	23	A496	1.5	0101
ABC301	2235	132.3	329.6	23	A302	2.5	0101

レコードタイプ

CHAR(6) CHAR(4) REAL REAL INT CHAR(4) REAL CHAR(4)

(3) フォーマット登録仕様

FGNO=20 <1H, 3X, 3A2, 2X, 2A2, 7X, 2F7.1, 7X, I4, 17X, 2A2, F6.1, 19X, 2A2>

注：アンダーライン部が各フィールドの出力フォーマットを規定する。フォーマットの書き方は、FORTRANのFORMAT文と同じである。

図6 FMS-Rファイル内容印字ユーティリティ出力例 ファイル内容印字ユーティリティでは、ファイル内のデータを、そのレコードタイプ、出力フォーマットに従って見やすく編集出力することができる。