

Title	ACOSシステム (4) : S700のソフトウェア その1
Author(s)	
Citation	大阪大学大型計算機センターニュース. 27 P.43-P.62
Issue Date	1977-11
Text Version	publisher
URL	http://hdl.handle.net/11094/65363
DOI	
rights	

Osaka University Knowledge Archive : OUKA

<https://ir.library.osaka-u.ac.jp/repo/ouka/all/>

ACOS システム (4)

— S700のソフトウェア：その1 —

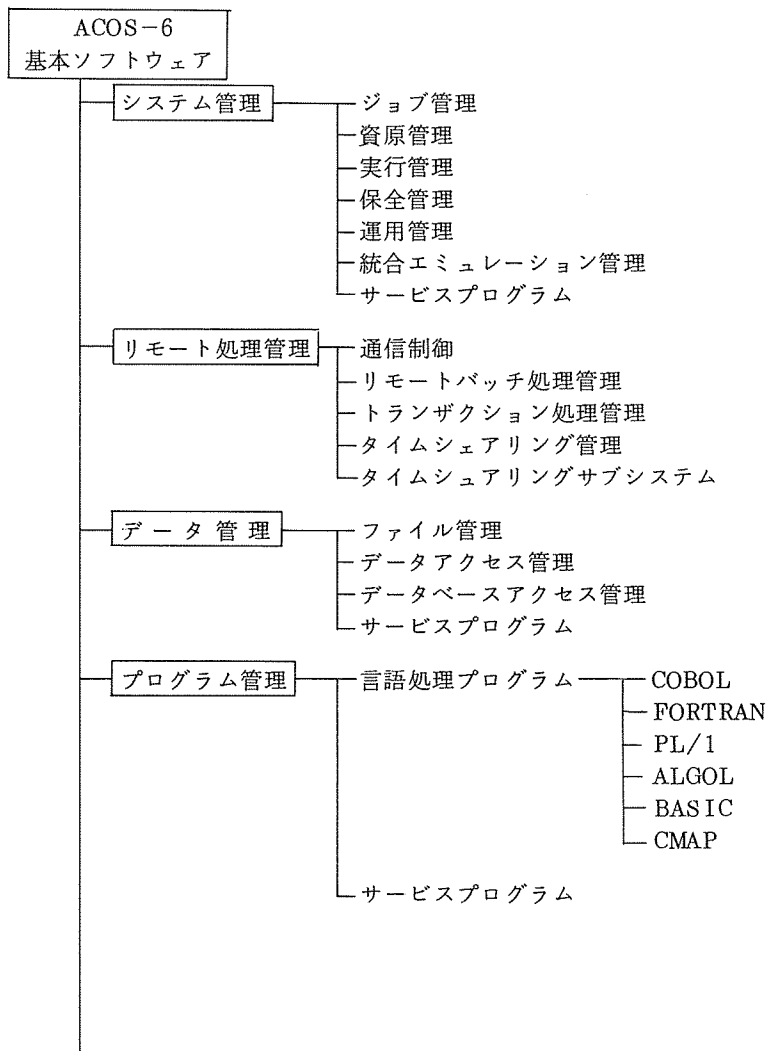
4. ソフトウェア：ACOS-6

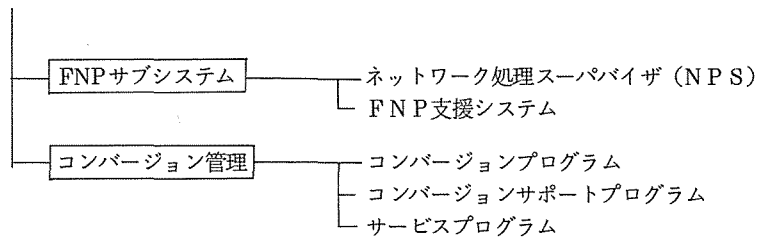
4.1 ACOS-6の概要

S700の中心思想であるISSを実現する機能としてデータベース機能、多次元処理機能、コミュニケーション機能、仮想記憶機能がACOS-6に具備されていることは、前に説明したが、これらの機能を実現しているACOS-6のソフトウェア体気を機能的に分類すると図4.1の様になる。

図4.1

ACOS-6基本ソフトウェア体系





次に、それぞれの機能について概要を説明する。

4.2 システム管理

システム管理はACOS-6の中核的な役割を果たすものであり、次のような機能をもっている。

- (1) ジョブの受け付け、実行および結果の出力（ジョブ管理）
- (2) 入出力処理やプログラムのロード、CPUの割り当てを始めとする種々のプログラムサービス（実行管理）
- (3) ジョブのアクティデティに対する資源の割り付け、解放および資源の管理（資源管理）
- (4) メモリの割り付け、スワッピング処理（資源管理）
- (5) ハードウェアシステムのエラー監視と回復（保全管理）
- (6) 操業員との交信、アカウント情報の収集および編集（運用管理）
- (7) システムの運用開始、中断、終了等の処理（運用管理）

これらの諸機能は、多重プロセジック、多量プログラミングおよび多重処理形態を提供し、システムの高性能性と高信頼性を実現するACOS-6の基礎となっているものである。

4.2.1 ジョブ管理

ジョブ管理は、ジョブの連続的処理を行うことを主要な機能としている。

システムインプットは、ジョブに関する記述指定の行われた入力ジョブの流れを読み取り、続く処理を考慮し、データとジョブ制御言語にそれぞれ分割してそれらをいったんシステムのディスク装置内へしまい込む（スタッキングの機能）、このようにスタックされたジョブは、システムスケジューラによって選択され（ジョブのスケジュール機能）資源管理のサービスをうける。

資源管理（メモリアロケータとベリフェラルアロケータ）は、各アクティビティごとに必要な資源の確保を行い実行すべき特定のアクティビティで利用される処理プログラムを呼び出し、管理の制御権を実行管理（特にディスパッチャ）に移す。処理プログラムの実行が完了すると、再びジョブ管理プログラム（ターミネータ）が機能する。つまりそのアクティビティに割り当てた入出力装置やメモリを他のジョブのために解放すると同時に、そのアクティビティで作成した。あるいは使用したファイルの後始末を行ったうえで、次のジョブまたはアクティビティの実行の準備にとりかかる。そしてジョブ

が終了した時点でシステムアウトプットが出力結果を印刷する。

図4-2は、一つのジョブがシステムの中でどのように流れていくかを示したものである。

この図に示されるように、一つのジョブは次の六つのシステムプログラムによって処理される。

(1) システムインプット

カード読取装置、磁気ディスク装置からジョブを読み込む。

(2) システムスケジューラ

システムインプットから読み込まれたジョブを実行してよいか否かの判断を下す。

(3) ペリフェラルアロケータ

主記憶以外の資源をアクティビティに割り当てる。

(4) メモリアロケータ

アクティビティに主記憶を割り当て、必要なプログラムをロードする。

(5) ターミネータ

プログラムの終了処理および資源の解放を行い、アカウントリングデータを収集する。

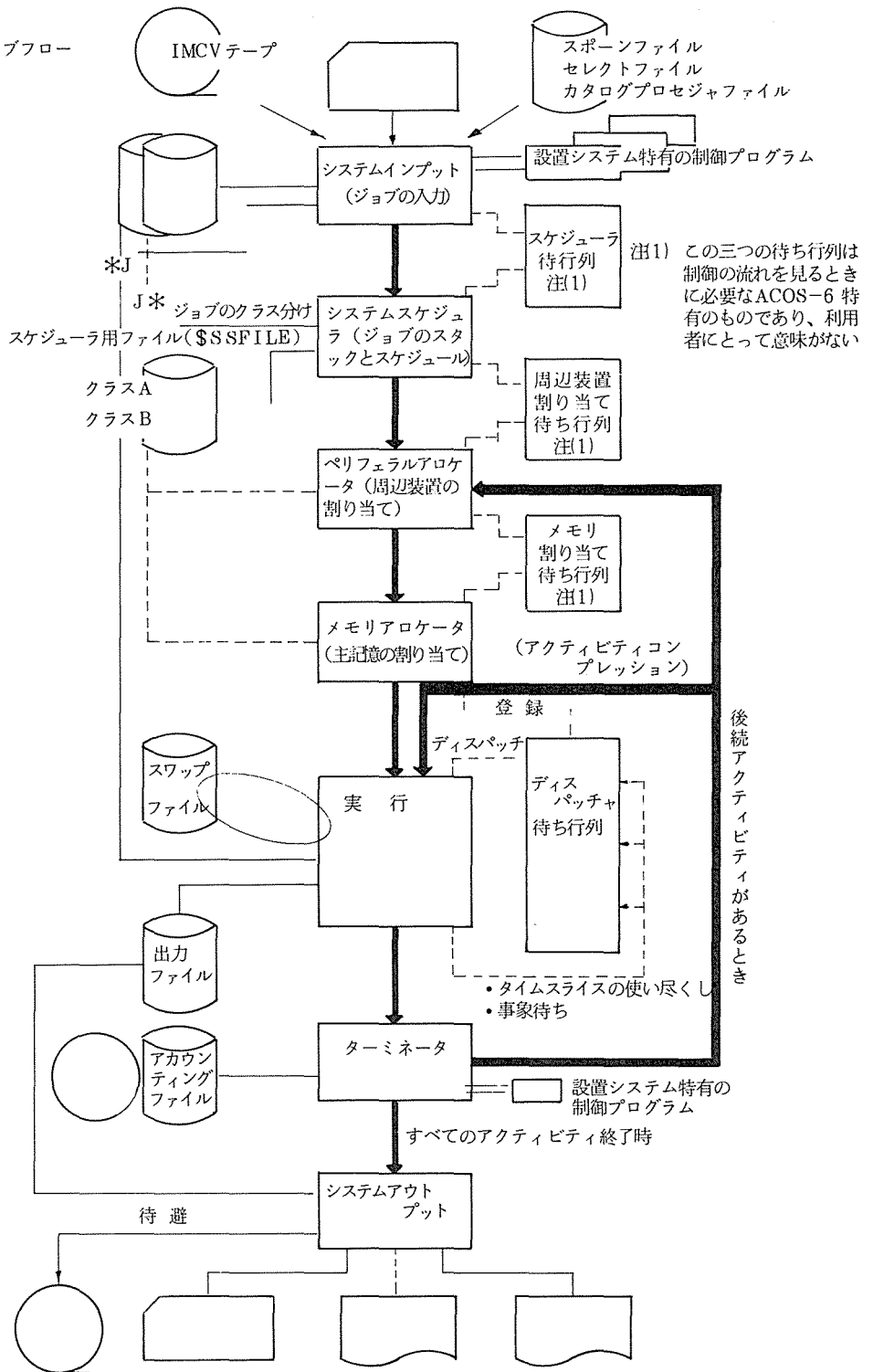
(6) システムアウトプット

プログラムによって出力されたデータをラインプリンタやカードせん孔装置や端末に出力する。

一つのジョブが複数のアクティビティより構成されるとき、(5)から(3)への復帰が繰り返される。ただし、あるアクティビティに続くアクティビティが前のものと全く同一の資源を利用している場合は(5)から実行に戻る。

この機能はアクティビティコンプレッションと呼ばれ、システムのオーバヘッドを減少させ、スループットを増加させる一因となる。

図4-2
ACOS-6 ジョブフロー



4.2.2 資源管理

情報処理システムの目標をむだなく効果的に達成するには、システムを構成しているハードウェア機能（資源）のすべてを可能な限り使うことが絶対条件である。ここで云う資源の代表的なものは、メモリおよび周辺装置であり、これらを集中的に管理している資源管理のコンポーネント（ペリフェラルアロケータとメモリアロケータ）は非常に重要な地位をもってくる。ここではACOS-6の資源管理の特長を紹介する。

(1) 優先処理

コンピュータ室の管理者が、管理者本来の仕事に没頭できるようにするためには、ジョブのスケジューリングとかデータの管理といったわずらわしいことからできる限り解放されている必要がある。とりわけ処理すべき作業（ジョブ）の実行順序の整理は、システムの性能が高ければ高い程それだけ負担は重くなり大変な作業となることは必至である。このためにも、処理すべきジョブに優先順位を示すコードを与えることができ、優先順位の高いものを優遇するような機構が必要である。ACOS-6では、次のような優先度並びに緊急度を設定し上記の要望に込えている。

- (a) スケジューリングの優先度
- (b) 資源割り当て時、実行時の緊急度
- (c) 出力時の緊急度

(a) スケジューリングの優先度

この順位を示すコードは、システムスケジューラが資源管理へジョブをゆだねるときの優先度を指す。これはシステムに蓄積されたジョブが多くあるとき、その内からどれを選び出すかというスケジューリングアルゴリズムに関連しジョブの特性（特質）とか順序関係を意識するとき有効に活用できる。図4-3にそのアルゴリズムを示します。

(b) ジョブの緊急度

この順位を示すコードは、資源の割り当てに関する順位を規定するもので次のものに対し影響を及ぼす。

周辺装置の割り当て

メモリの割り当て

CPU時間の割り当て

出力結果の出力装置割り当て

このパラメータは、ジョブの頭に付けるJCL（\$SNUMB文）にセットすることができ、利用者が特に急ぐジョブをもち込んだとき、もしくはレスポンスタイムを気にするアプリケーションに利用すると効果が表れる。

(2) 作業用ファイルの確保

利用者が使用するファイルには次の二つがある。

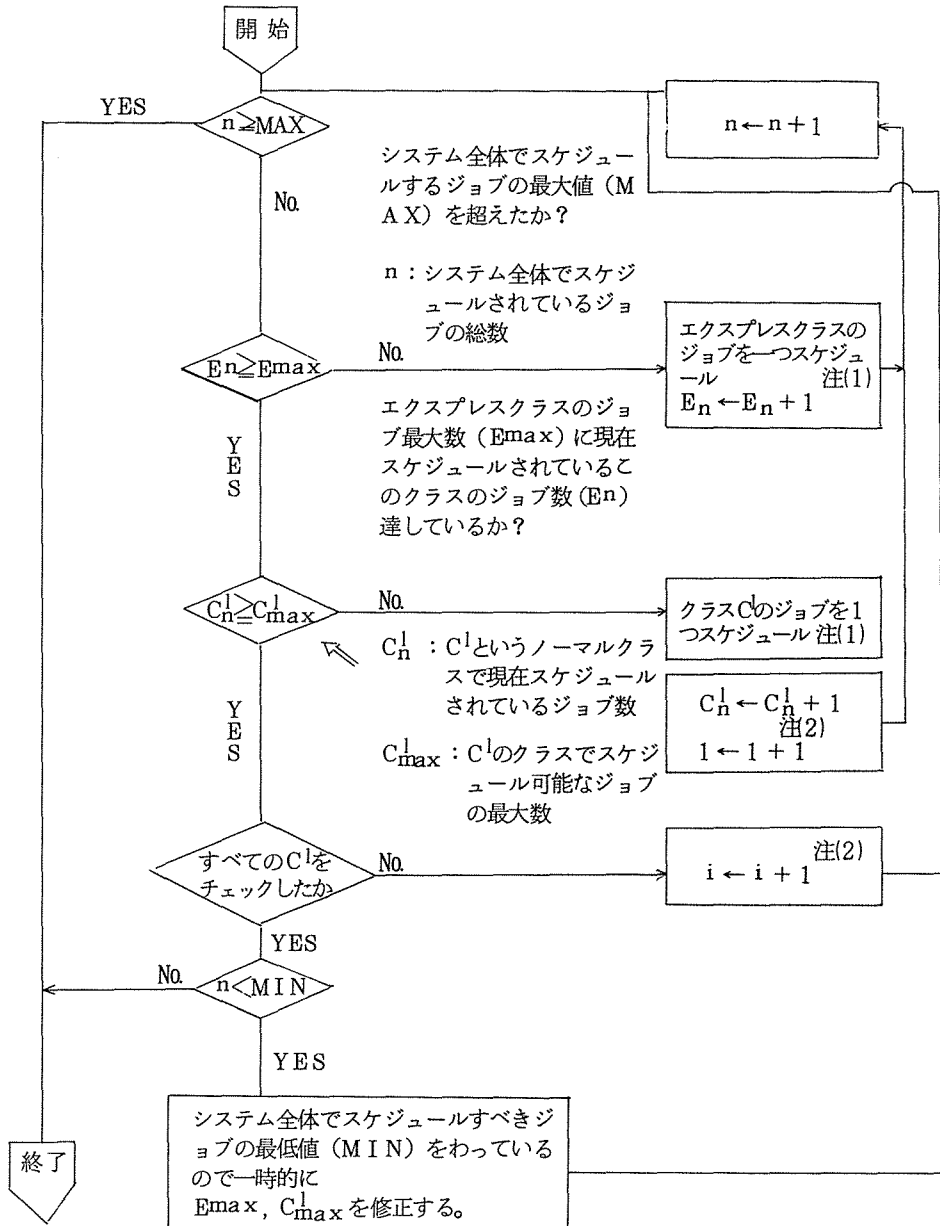
- (a) 一時的に使用しすぐに不用となる作業ファイル
- (b) 半永久的に使用されるパーマネントファイル

この両者はオペレーティングシステム上での扱いも完全に区別されるべきであり、特に作業用ファイルの確保、解放をパーマネントファイルのそれと区別しなければ、運用上大変な支障をきたす。

このためACOS-6では、\$FILEというカード一枚でこの作業用ファイルをディスク上に自動確保し、使用后自動解放することが可能である。このとき確保されたファイルスペースが実行中不足した場合でもスペースの自動追加がなされるので、プログラムの余計な心配をなくさせると同時に、オペレータも運用管理者もファイルスペースの管理等の作業を大幅に縮小させることができる。

図 4-3 ジョブのスケジューリング

アルゴリズム



- 注(1) 優先度に差があれば、優先度順にスケジュール
- (2) 全クラス数に達したときはまた 1 に戻す。
- (3) 上図中のコンスタントはスタートアップ時に決定される。
又運用中に操作卓より変更できる。

4.2.3 実行管理

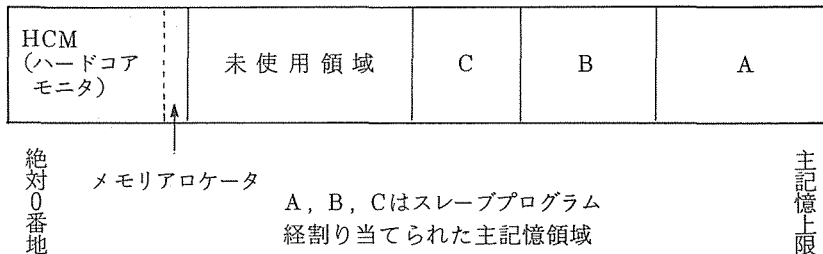
ジョブ内の一アクティビティに対し、資源の割り当てが完了するとプログラムが主記憶にロードされ、ディスパッチャと呼ばれるモジュールの制御下に入り利用者プログラムが働きはじめる。ACOS-6はプログラム実行中に種々のサービスを行うモジュールを用意している。これらのモジュールは、ソフトウェア体系の中ではシステム管理下の実行管理に属す、この節で実行管理の概要を説明するが、システム管理の中核は実行管理であり、スタートアップを除くすべてのプログラムは実行管理なくしては動作不可能であることに留意されたい。

実行管理を構成するものとして、次のものが挙げられる。

- (1) ディスパッチャ
- (2) 入出力スーパーバイザ
- (3) フォルトプロセッサ
- (4) MMEプロセッサ (Master Mode Entry)
- (5) 入出力例外プロセッサ

図4-4は、ACOS-6の主記憶のレイアウトを示すものである。

図4-4 ACOS-6主記憶レイアウト



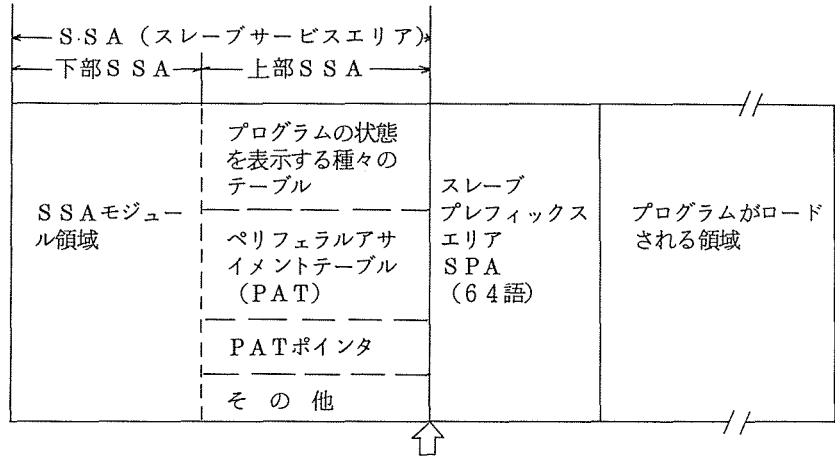
前述の実行管理を構成するモジュールの多くは、このHCM内に実装されている。

そして、主記憶全体からHCMを除く残りの部分はすべてのスレーブプログラムに割り当てられる領域である。HCMの大きさはシステムの構成によって可変であり、また、設置システムで準備したコンポーネントの追加も簡単である。図4-5は、一つのアクティビティに対して割り当てられる主記憶領域を拡大して見せたものである。

図4-5からもわかるように、各アクティビティに対してSSAと呼ばれる領域が割り当てられ、SSAモジュールはこの中にロードされる。SSAモジュールとは、おのこのアクティビティ(スレーブプログラム)が固有に要求する種々のサービスを行うものである。従って、システム全体から見てあまりひんぱんに利用されないモジュールは、このSSAモジュールとして実装される。MMEプロセッサの一部とか入出力例外プロセッサの大部分は、SSAモジュールとして実装されている。このように、SSAの特色を利用しながらシステムを実装する方法を分散モニタ方式と呼ぶ(distributedmpnitor)

備考 ある設置システムで特定のSSAモジュールがひんばんに利用されることが考えられるときは、スタートアップ時の指定によりHCM内へ簡単に実装できる。

図4-5 アクティビティに対し割り当てられる主記憶領域



相対0番地 : BAR (ベースアドレスレジスタ) はここを指す。

SSA (スレーブサービスエリア) : 通常1K語の大きさを持ち、アクティビティの固有情報をもつ。

下部SSA : 512語の大きさを持ち、SSAモジュールがロード実行される。

上部SSA : プログラム (アクティビティ) の状態を表示する種々のテーブルとアクティビティに対してどのような周辺装置やファイルが割り当てられているかを表示するPATとPATポインタをもつ。

SPA (スレーブプレフィックスエリア) : ACOS-6とプログラムとの通信領域である。

SSAの大きさは、次のように決定される。

- (1) 通常のジョブ \$LIMITS文で2.5K以上のプリント行数を指定 : 2K語、そうでないとき : 1K語
- (2) TSE 同時アクティブ端末数が18まで3SSA、以上18増えるごとに1SSA追加 (最高18SSA)
- (3) TPE 通常2K語

ここで実行管理の代表的機能であるディスパッチャについて説明する。

ACOS-6は、最大4台までの中央処理装置を制御することが可能である。ACOS-6が提供する多量プロセッサシステムは、主記憶を中心としたシステムであるから、従来のシステムに中央処理装置だけを付加したり、あるいは故障時に中央処理装置を切り離すことは簡単である。これはハードウェアのモジュール化を行い、コストパフォーマンスを極限にまで追求したことの一つの成果である（すなわち、中央処理装置は演算能力と関連するハードウェア装置であり、それ以外の機能例えば主記憶の制御等に関連するハードウェアをもたない）

二つ以上の中央処理装置が存在する場合、システムは各中央処理装置にかかる負荷が常に平等であるように制御される。従って、一つのアクティビティが t_1 という瞬間ではAという中央処理装置でサービスされているとしても、次の t_2 という瞬間において同一の中央処理装置でサービスされているという保証は一般にない。各中央処理装置は、ディスパッチャと呼ばれるシステムプログラムを走るとき、自分が実行すべきプログラム（アクティビティ）を選択する。

このディスパッチャがもつ機能をまとめると次のようになる。

- (1) 各CPUに対し、次にサービスすべきアクティビティを教える。
- (2) ある事象が発生するまで、そのアクティビティのサービスを中断する。

事象としては、すべての入出力要求が完了すること。

入出力要求のいずれか一つが完了すること。

ある時間が経過すること（注1）

他のプログラムが呼び出すこと等がある。

- (3) アクティビティがレディになったとき、CPUのサービスを受ける候補とする。
- (4) 多重プロセッサシステムにおけるCPU相互干渉をなくす。

（注1）待ち合せ時間を3秒以上にすれば、スワップアウトの候補となる通常一つのスレーブプログラムが連続的に使用できるCPU時間の最大値は $96 \text{ ミリ秒} \times (\text{CPU台数} + 2) \times 32 \text{ ms}$ ）である。

4.2.4 運用管理

ACOS-6では、パーティション（システム生成時に決定されるジョブを実行させるために設けられたメモリ上の区画）という概念がないので、メモリの有効利用が完全に行われる反面自由にジョブが投入できることから、従来に比べてジョブの投入、スケジュール、出力に関する有効な機能が必須になっている。そして多重プログラミング方式では、多くのジョブが同時処理されることからシングル処理と比べてオペレータの仕事は、複雑にならざるを得ない。

また、タイプライタ型の操作卓では刻々と変化するシステム全体を一目で見渡すことは困難である。もしテレビ画面のような状況表示装置を通して、マルチプログラミングを監視する機能があれば、オペレータの負担はずっと軽くなる筈である。

さらに磁気テープのかけ替え、ディスクパックの交換を間違えるとジョブが異常終了するだけでなくデータまでも破壊されてしまうのでは、オペレータは相当に慎重にならざるを得ない。そしてオペレーションの間違いを恐れる余り多重プログラミングがシングル処理と余り変わらなくなってしまう危険性もある。この意味で多重プログラミングの真の効果を得るためには、システムの運用管理者の業務および、オペレータの操作を容易にするための機能が重要な位置を占めてくる。

A C O S - 6 では、次のような機能を完備しシステムに対する要望に応えている。

- (1) ジョブのスタッキング機能 (ジョブ管理)
- (2) 自動ボリューム認識機能 (資源管理)
- (3) カタログプロセッサ機能 (ジョブ管理)
- (4) ファイルのカタログ機能およびファイル世代管理機能 (データ管理)
- (5) システムの構成、再構成を簡単に行うシステム生成機能 (スタートアップ)
- (6) システムを稼働させるための初期化機能 (スタートアップ)
- (7) システムを運転、維持していくための操作員との交信機能 (オペレータインタフェース)
- (8) 運転状況を把握し、効率よく稼働させるためのシステム状況表示機能 (VIDEO)
- (9) コンピュータ室を合理的に運用していくためのアカウントिंग機能
- (10) 運転効率化のための各種資源活用状況の把握と分析機能 (サマリエデットプログラム)
- (11) 障害原因の追跡システム評価のために有効な情報を与えるトレース機能

このうち運用管理上重要な機能として操作員インタフェース機能を以下に説明する。

A C O S - 6 と操作員との交信は、システム操作卓 (SCC: system control center) を通して行われる。システムは、最大 5 個のディスプレイを制御し多重操作卓機能を提供する。また、装置故障に対しては他の操作卓をもって代替する。各操作卓には TY1 ~ TY5 の装置名が割り当てられるが、TY1 ~ TY4 を一つの操作卓に割り当てることも可能で、この場合、メッセージは一つの操作卓にまとめられる。TY5 は一般にシステムの動作状況を表示するプログラム V I D E O (図 4 - 6 参照) のために利用される。(ステータスディスプレイ) TY1 ~ TY4 に対し、出力されるメッセージは次のとおりである。

TY1

主操作員用の操作卓であるとされ、ジョブの実行、終了に関するメッセージとかエラー報告が出力される。

TY2

磁気テープ、ディスク関連メッセージ

例: 磁気テープや磁気ディスクパックのマウント要求

TY 3

ユニットレコード装置関係メッセージ

例：ラインプリンタの紙切れとかカードせん孔装置に対する処置を指令

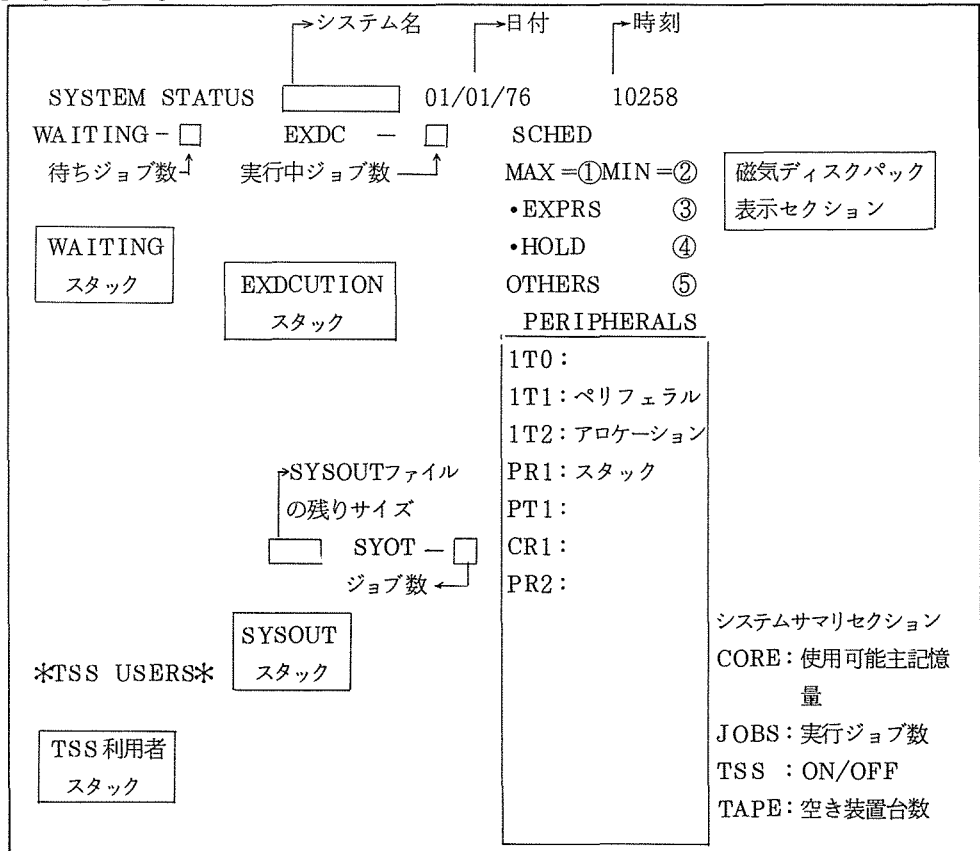
TY 4

媒体要求メッセージ

例：特定の名前のついた磁気テープリール等を必要とするとき出力されるメッセージ

VIDEOは、次に示す画面をステータスディスプレイ等に表示するプログラムである。

図4-6 VIDEO



4.3 データ管理

今日の電子計算機システムにおいて、複雑な論理関係をもつ多種多様なデータの統合的管理と共用の重要性はあらためて述べるまでもない。この大きな命題をACOS-6のデータ管理は実現しており、その概要は以下に述べるとおりである。

ACOS-6のデータ管理は制御の対象となるすべてのデータやファイルを統合的に管理し、かつ高度な操作性と完全な機密保護および保全能力を持つソフトウェアシステムか

ら構成されている。これらのシステムにより情報の集約化と一元化が可能となるので、多数の利用者があらゆる利用環境から相互に情報を共用し有効に利用することができる。また利用者が情報を安心して利用できるようにするために、利用者、ソフトウェアまたはハードウェアの故意または偶然によるあらゆる障害から情報を保護している。

データ管理は機能的に以下のような各形態に分かれており、図4-7に示すように互いに密接な関係で結ばれ効果的な情報処理手段と便宜を提供している。

- (1) ファイル管理 ファイル管理システムFMSはファイルの作成、管理、維持と、侵略や障害および共用による相互干渉に対するデータの保全と機密保護を行う。
- (2) データアクセス管理 統合ファイルアクセスシステムUFASは種々のファイル形態に対し、適切なアクセス方式により効果的な入出力制御を行う。
- (3) データベース管理 情報の統合と一元化を可能にしたデータベースシステムIDSに加え、CODASYL データベース仕様に準拠した本格的なはん用統合データベースシステムADBSがあり、効果的でしかも使いやすい情報管理手段を提供している。さらにデータベースに対する高度のスキルを持たない人でも、データベースの作成/使用が容易にでき、非定型な検索が効率よく高速にできるデータベースシステムINOもあり、環境、目的、スキルに合ったデータベースシステムを選ぶことができます。
- (4) サービスプログラム データの分類や併合および媒体変換などのサービス機能を行う数多くのサービスプログラムが用意されている。

これらのデータ管理ソフトウェア群はオンライン機能とも密接に結びついており、オンラインによる強力なデータベースの管理とその処理が可能となっている。

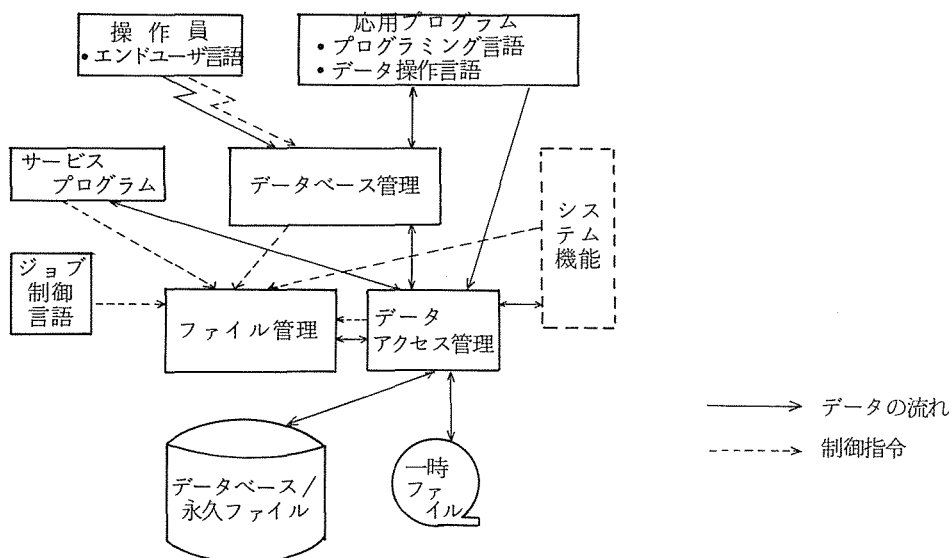


図4-7 データ管理

4.3.1 ファイル管理

すべての永久ファイルはカタログとファイルから成る階層構造になっておりFMSによって統一的に管理される。利用者はあらゆる処理状態からこの永久ファイルを相互に共用し利用することができる。FMSはファイルを管理し維持するために、カタログやファイルの作成、削除、変更などのカタログ操作と、大容量記憶装置のスペース制御、さらにファイルの保全と機密保護を行っている。

ファイルの保護としては、第一に権限のない利用者による故意または誤ったファイルのアクセスを防ぐため、機密語、パーミッションおよび機密錠による万全の機密保護機能を備えている。第二にハードウェア機器の誤動作や障害からファイルを守るために、書き込み確認、ファイルの二重化、活動記録およびファイルの復元などのファイル保護と回復機能をもっている。第三に利用者プログラムの誤操作に対する保護として、錠、ロールバック、チェックポイント/リスタートおよび遅れ処理機能が用意されている。最後にファイルを共用し同時に並行して実行するアクティビティ間の相互干渉に対する保護のためには、並行アクセス制御と同時更新制御がFMSにより行われている。(図4-9)

4.3.2 データアクセス管理

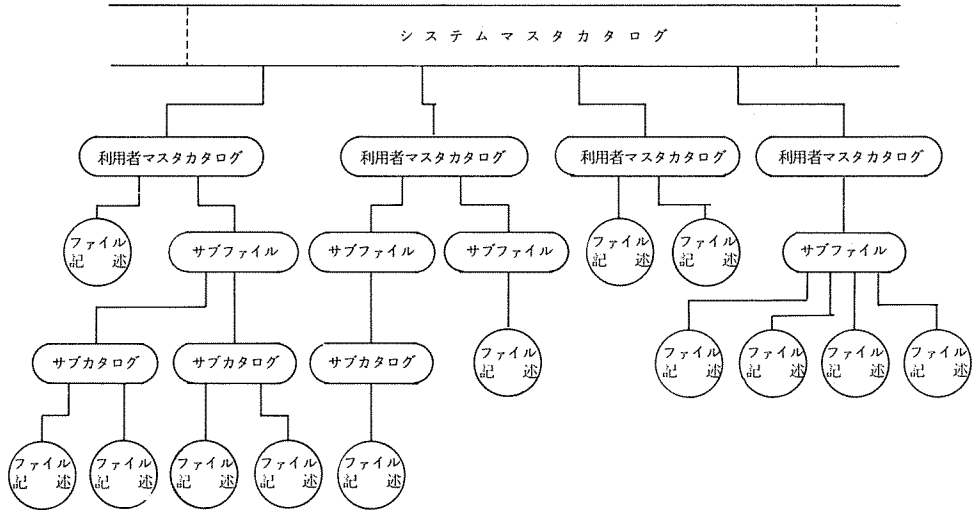
UFASは処理の対象とするあらゆるファイルの論理および物理入出力を統合的に管理制御し、各種のファイルに共通な論理レコードインタフェースを用意することによって、簡単にすべてのファイルを処理することができるようになっている。このようにはん川で効果的な入出力制御を達成しているUFASの主な基本機能を列挙すると次のようになる。

- (1) ファイル形式とラベルの処理
- (2) バッファ管理と共用制御
- (3) レコードの物理的/論理的処理
- (4) レコードのブロッキングとバッファリング
- (5) エラー検出と回復処理

UFASはシステム標準形式ファイルのほかに、IBM・EBCDIC磁気テープファイルやANSI標準形式ファイルなど広範囲な各種形式のファイルも混合して取り扱うことができる。

ファイル編成の種類には、順編成、相対編成、索引順編成および統合編成があり、アクセス方式として各編成に対応した順アクセス、相対アクセス、索引順アクセス、統合アクセスおよび直接アクセスが用意されている。

図 4-8 カタログ/ファイルの構造



備考 利用者マスタカタログ（UMC）に直属するファイルを特にクイックアクセスファイルと呼ぶ。

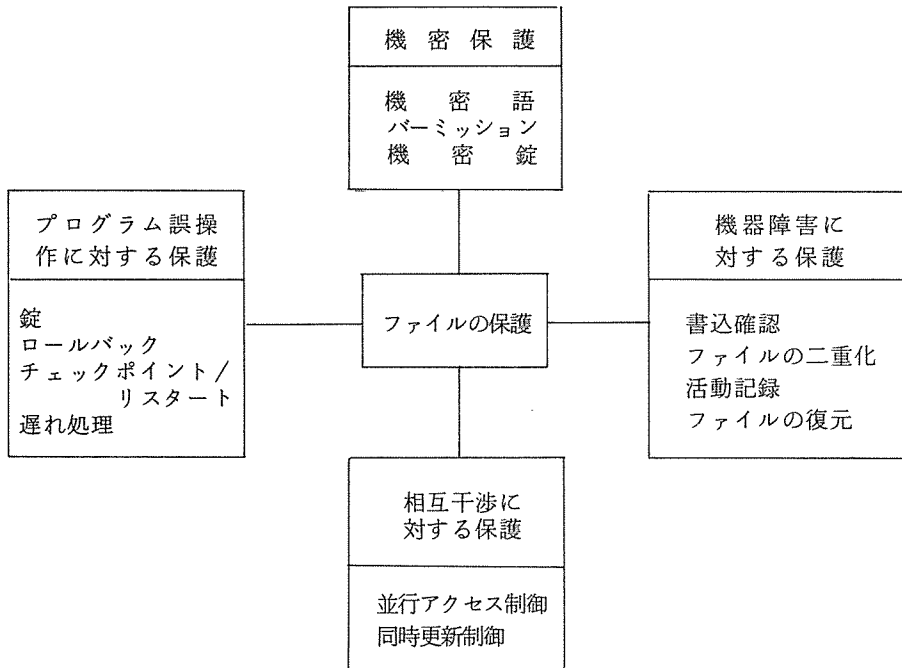


図 4-9 ファイルの保護

4.3.3 データベース管理

(1) INQ (INformation Query)

INQはACOSを用いてデータベースを構築し、情報システムを実現しようとするシステム設計者に提供されるデータベース・マネジメント・システムである。INQは、アプリケーション・システムに対して次の様な適用性がある。

- シーケンシャル・ファイルを主とする従来からの大量のファイル・マネジメントシステムから移行が容易である。
- 従来システムの拡張という発想でシステムの具体化がすすめられる。
- 従来からのファイルという観念を、データベース化に当り全て捨て去るのは容易でない。この「ファイル」という考え方のまま、データもしくはデータベースが扱える。
- 最大の特徴は、複雑な条件検索に対する高速の応答性にある。非定形（アドホック：adhoc）な問合せ、検索が主となる情報システム化に適する。
- 情報システムに対して、外部の変動要因は常に影響を与える。データベースの変更に対して常に柔軟に対処できる。システム設計者はデータベース変更に対して困難さを感じない。例えば、新しいフィールドの追加は、データベースに格納されているデータに、手をつける事なしに実現される。
- 個別に存在する2つのファイルをデータベース中で目的に応じダイナミックに結合させ、1つのファイルとして用いることができる。このファイル結合による効果はチェーン型データベースが持つ特徴であるファイル間のデータ構造に対応する効果を生む。しかも、この「構造」は動的に定義される。
- データ格納上、本質的に不必要な部分、例えばデータのないフィールド、フィールド中の空白や零を格納せず、データベースの格納効率を高める。
- データとして、不定長、不定形なもの、例えば文章情報を効率的に取り扱うための機能が用意される。

さらに、INQは新言語としてCOBOLだけではなくFORTRANからも使用できるため、二次処理として高度な計算処理も行なうことができます。

又、TSSモード下で、EUL (End User Language) により、会話的にデータベースの高度な検索/加工が図4-10の様になります。

図4-10 EULによる検索例

```

$$$ CON, TSS
ACOS-6 TSS(R3.0) ON 06/01/76 AT 16.552 CHANNEL 0102
USER ID-ACOS
PASSWORD-
#####
SYSTEM? INQ
OPTION FILE ? Cn
INQ SECTION? JINJI, INQSFIL
INQ FILE? JINJI-FIL
PASSWORD (JINJI-FIL) ? #####
INQ CAT/FILE? JFILO1
? RETRIEVE/ シュシチ = ▼オオサカ▼ AND シンチョウ > 170/
      3 RECORD FOUND
? SORT/ニューシャネン ASC/
? DISPLAY/シャイン# ナマエ カゾク/
      シャイン#      ナ マ エ      カ ゾ ク
      0 0 7      アオキ シゲル      アオキ ヨウコ
                                   アオキ イチロウ
                                   アオキ ハナコ
      0 1 1      タナカ タロウ      タナカ アキコ
                                   タナカ シンタロウ
      0 1 6      ナカムラ ジロウ      ナカムラ ヨシコ
? SAVE 1
? RETRIEVE/ キホンキュウ > 150000 /
      15 RECORD FOUND
? SAVE 2
? AND 1, 2
      1 RECORD FOUND
? DISPLAY / シャイン# ナマエ /
      シャイン#      ナ マ エ
      0 0 7      アオキ シゲル
    
```

```

? TABLE/シンチョウ BET 0 150 160 170 999 /
/ シュシチ EQ ▼オオサカ▼ ▼トウキョウ▼ /
シンチョウ
X1 : 000CM - 150CM
X2 : 150CM - 160CM
X3 : 160CM - 170CM
X4 : 170CM - 999CM
シュシチ
Y1 : オオサカ
Y2 : トウキョウ
      X1 X2 X3 X4
Y1 15 18 130 121
Y2 5 63 145 112
? MACRO RETR 1
?? RETRIEVE/ シュシチ = %A AND シンチョウ
?? SORT / ニューシャネン ASC /
?? DISPLAY / シャイン# %C /
?? Cn
? CALL RETR1, %A=▼トウキョウ▼ %B=160 %C=
      3 RECORD FOUND
      シャイン#      ナ マ エ
      0 0 8      カワシマ トウル
      0 1 2      キムラ ヨシオ
      0 1 4      ササキ シゲル
? DONE
      INQ DATA BASE RETRIEVE END
SYSTEM? BYE
** COST: ¥110 TO DATE ¥32,145 = 0%
** ON AT 16.552 - OFF AT 16.772 ON 06/01/76
    
```

(2) 統合データベース (IDS)

IDS (integrated data store)は、ACOS-6のもとで動作する統合データベース管理システムである。これにより、大量かつ組織的なデータを処理するシステムの作成・利用・保守およびプログラミングに要する費用を減少させることができる。

IDSは、大容量のランダムアクセス記憶装置を主記憶装置の拡張部として用い、効果的なデータ編成手法を提供すると同時に、単純かつ効果的な言語をプログラマに提供する。従来のファイル管理システムは、フィールド単位の処理やシーケンシャルなレコード処理に適していたが、大容量ランダムアクセス記憶装置を用いたランダムレコード処理には適していなかった。しかし、IDSのCOBOL形式のIDS言語を用いれば、組織的な(構造的な)レコード処理が簡単に行えるようになる。

IDSにより扱われるデータベースはIDSファイルと呼ばれ、64個までのエリアに分割可能である。更にエリアは、サブファイルに分割される。このサブファイルはFMSでの1ファイルに対応している。一つのプログラムは、すべてのエリアをアクセスすることが可能であり、最大644億バイトの大きさをもつデータベースが扱われる。また多重アクセスIDSは、一つのデータベースに対し、二つ以上の利用者プログラムがアクセスすることを許容している。

IDSデータキューリを利用すれば、同時に多数の端末からオンラインIDSデータベースを検索できる。

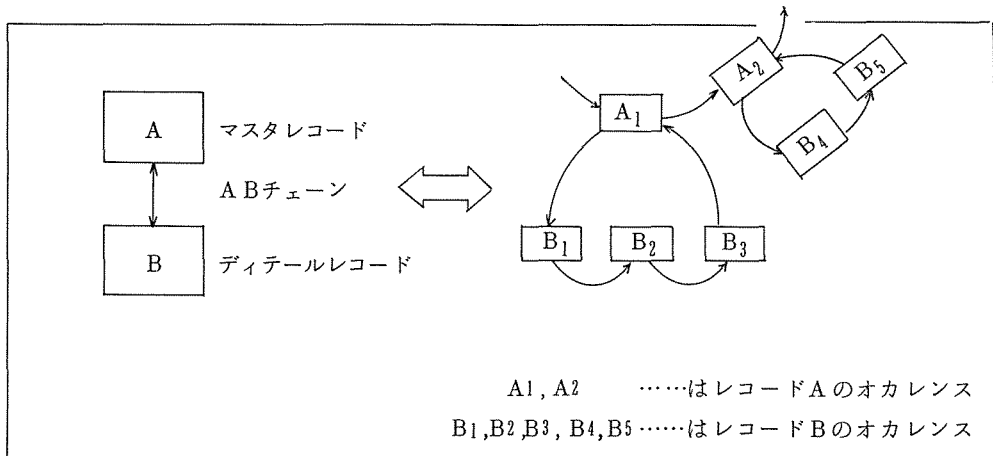
また、IDSはCOBOLを親言語にしているので、COBOL自身も豊富な機能を十分に利用できる。この節ではIDSの

- データ構造
- 記憶構造
- 格納方法と検索方法
- 多重アクセス

が説明される。

データ構造とは、レコードの論理的な構造(関係)である。

IDSにおいて、レコードとはデータの格納、データの検索、データの削除等の単位であり、二つのレコードを考えると一般にマスタとディテールという相対的な関係が存在する。マスタレコードとディテールレコードの一组は、一つのチェーンを構成し、幾つかの構造をとる。最も基本的な構造は、一つのマスタレコード(A)が一つのディテールレコード(B)をもち、ABチェーンが構成されているものである。



(3) 統合データベース (ADBS)

ADBS (Advanced Data Base System) は ACOS-6 のもとで統合データベースを設計し、生成、運用する強力な機能をもつデータベース管理システム (DBMS …… Data Base Management System) である。

ADBS を使うことによって複数のアプリケーションに共用される情報を含む統合データベースを構築することができ、このデータベースに対するいろいろな処理要求に対処することができる。

大容量ランダムアクセス記憶装置にデータを格納したり、検索したりするデータ操作機能は、ADBS 用に拡張された COBOL 言語 (COBOL74) を使うことによって簡単に行うことができる。

ADBS は ACOS-6 の DBMS の一つである IDS がもつ機能に加えていくらかの新しい機能をもっているので、より大規模で豊富な機能をもつデータベースシステムを構築することができる。これらの新しい機能はデータベースの設計に柔軟性を与えるとともに CODASYL の DBMS 標準化提案に準拠しているので将来への発展性をもつものである。

ADBS はデータベースシステムの設計と運用に柔軟性を与えるために以下にあげる豊富な特長をもっている。

(1) データの独立性

ADBS はデータベースの構造記述をプログラムから分離しているのでデータベースの構造が変わってもそのデータベースを使っているプログラムの再コンパイルは必要ない。この機能によってデータベースの拡張や変更が容易になる。

(2) データ構造の多様性

ADBS は適応領域の広範な要求に対処するために、いろいろな種類のデータ構造を提供できる。具体的には単一構造、トリー構造、ネットワーク構造およびそれらの自由

な組み合わせが可能である。

(3) 大規模データベース

A D B Sでは最大680億レコードを含むデータベースを作ることができる。

(4) 保 全 性

A D B Sはデータベースの保全性を維持するためにシステムジャーナルの収集や自動的なリカバリ/リスタート機能を提供するとともに、レコード関係を表現するポインタのシステムによる矛盾のない維持が行われている。またユーザが利用できるデータベースの部分を制限することによっても保全性を実現している。

(5) データの機密保護

プライバシロックとキーを使ってA D B Sはデータベースに対するアクセスをデータアイテムのレベルまで制限することができる。

(6) ハードウェアからの独立性

データベースシステムの将来への拡張性を容易にするためにA D B Sはプロセッサおよび装置からの独立性を実現している。

(7) 記憶スペースの削減

冗長データを排除したり、レコードが削除されたときにスペースコンパクションを行うので記憶スペースが削減されその費用とファイル維持作業を減少することができる。

(8) 再構成ユーティリティ

広範な再構成機能によってデータベースの拡張または変更要求に対して容易に対処できる。