

対話型システムとしてのVM/370 CMSの諸機能

小南秀一郎 (日本アイ・ビー・エム)

1. はじめに

VM/370は1台のハードウェア上で複数の仮想計算機を稼働させる機能を提供すると共に、その仮想計算機上で動く対話型OSとしてCMS (Conversational Monitor System) を提供している。ここではCMSの提供する機能について説明すると共に複数のCMS間のコミュニケーションのためにVM/370のCP (Control Program) が提供する機能について説明する。

2. CMSの提供する機能

2.1 CMSの構成

CMSがサポートする仮想計算機の最大構成は図1に示すとおりである。

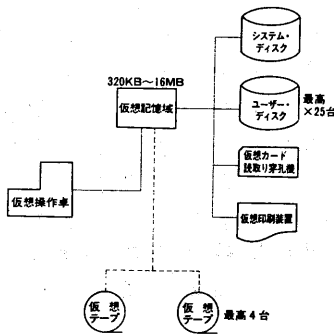


図1 CMSの最大構成

ディスクは実際のディスク・スペースを分割し別々の仮想計算機に割り当てるミニ・ディスク (CPが提供する機能) を使用する。図1に示す以外の装置もユーザーがSIOレベルのコーディングを行えば接続可能である。

2.2 CMSのファイル

CMSミニ・ディスク上のファイルは使用する前にFormatコマンドによ

り1K, 2K, 4Kまたは800バイトにPre-Formatされて使用される。ファイルを構成するブロックはポインターによりチェーンングされ空きスペースはアロケーション・マップにより管理される。従ってユーザーは個々のファイル作成時にスペースを指定する必要がない、またミニ・ディスク全体の再編成等の必要もない。固定長および可変長レコードに対する順次および直接 (レコード番号による) アクセスが可能である。

CMSファイルはファイル名、ファイル・タイプ、ファイル・モードにより一意に識別される。ファイル・モードはAからZの値を取り、そのファイルが入っているミニ・ディスクを示す。ミニ・ディスクはAccessコマンドによりアクセス可能になるがその時指定したモードによりその時点のミニ・ディスクのモードが決る。ファイル・モードを指定しない場合の検索順はA→Zである。ファイル・タイプは通常ファイルの用途を表すのに用いられる。特定のプロセッサは特定のファイル・タイプを要求することがある。例えばAPLのワークスペースはファイル・タイプVSAPLWSとなる。ファイル名はいわゆるファイルの名前として使用される。

複数の仮想計算機から同一ミニ・ディスクをアクセスすることが出来る。これはCPの提供するLINKコマンドによるが、このとき読取のみ、読取/書込、1ユーザーのみ書込の3種のパスワードによりミニ・ディスクへのアクセスをコントロールできる。

2.3 CMSの記憶域割当て

CMSの記憶域割当てを図2に示す。

記憶域 X'20000' までは主に CMS の Nucleus ルーチンが使用する。通常ユーザー・プログラムは X'20000' からロードされるがプログラム作成時の指定により X'E000' から X'10000' の間にもロードできる。CMS が提供するコマンドのサービス・ルーチンにはこの2つのエリアで実行されるものもある。ロードされたプログラムのエンタリー・ポイント等を保持するテーブルは高アドレスからとられる。Access コマンドによりアクセス可能になったミニ・ディスク上のファイルに対するポインターも高アドレスからとられる。

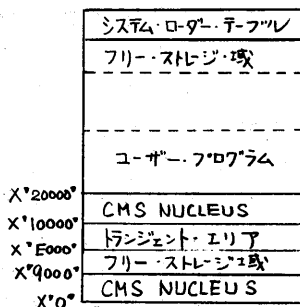


図2 CMS の記憶域割当て

2.4 DCSS (DisContiguous Saved Segment)

複数ユーザーが共用するルーチンはユーザー毎にコピーを持つのではなく1個のコピーを記憶域に常駐させこれを共用して記憶域の有効利用をはかることは通常のOSにおいてもよく行われている。CMSにおいてもエディター、EXECプロセッサ等CMSのルーチンのみでなくAPL、GDDM (Graphical Data Display Manager) 等ユーザー向言語、グラフィック・ルーチンに複数ユーザーで共用するものが多数存在する。しかしCMSは1ユーザーに対し1仮想計算機を対応させるため複数計算機から記憶装置上の1コピーをAddressableにするために工夫を要する。CMSはCPの提供する

DCSS機能を用いてこれを行っている。

DCSSを作成するにはまずCPのテーブルにその名前、記憶域にロードされたときのスタート/エンド・アドレス等を指定した後ディスク上の保管域に保管する。ディスク上の保管域には記憶域にロードされたイメージで4Kバイトのページ単位に保管される。

ユーザーがDCSSを使用する場合にはその名前を指定してCPに要求を出す。CPは(そのDCSSがまだ記憶域にロードされていないければこれをロードした後)DCSSのアドレスに対応するページ・テーブル、セグメント・テーブルをそのユーザーのために作成する。これによりこのユーザーから要求したDCSSがS/370のDAT (Dynamic Address Translation) 機構を通じてアドレス可能となる。(図3)。

DCSSに関するサービスをCPに要求するにはDiagnoseインストラクション(オペレーション・コードX'83')が使用されている。このコードは標準のS/370のインストラクションセットではハードウェア・モデル特有のサービスを要求するコードとして定義されているがVM/370においては仮想計算機がCPに種々のサービスを要求するため使用されている。

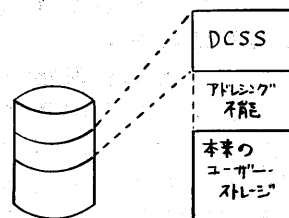


図3 DCSS

2.5 CMSのSVC処理

CMSに対するサービス要求のためにSVC(SuperVIsor Call)インストラクションのコード202, 203が使用される。これはOS, DOS系のSVCコードの使用法とつぎの2点で異っている。

- OS, DOS系においてはSVCのコード0から199をシステムのスタンダードなSVCとして使用し200以上はユーザーSVCとしてユーザーに開放されているのに対し, CMSでは202, 203をシステムのSVCコードとして使用する。
- OS, DOS系においては要求するサービスに対応してSVCコードを割振っているため100種を超えるコードがシステムのコードとして使用されているのに対し, CMSではシステムのコードとして202, 203のみが定義されている。

一 CMSはOS, DOS系のSVCに対してシミュレーション・ルーチンを提供している。SVCコード0から199のOS, DOS SVCコードをCMSが受けるとCMSはシミュレーション・ルーチンにコントロールを渡す。OS, DOSいずれのシミュレーション・ルーチンにコントロールを渡すかはCMS・Nucleus中のビットのオン/オフによるが, このビットを操作するためのコマンドが用意されている。この方法によりCMS上でOS, DOS系のほとんど全てのコンパイラを動かすことが可能であり, またOS, DOS系の主要なアクセス・メソッドをシミュレートすることによりこれらのアクセス・メソッドを使用しているプログラマを実行することができる。

OS, DOSのシミュレーション・ルーチンは大部分が2.4項で説明したDCSSとして提供される。

2.6 CMS EXEC

CMSはEXECおよびEXEC2と称されるコマンド・リスト機能を提供する。共にCP, CMSコマンドの他に変数のアサイン, 置換, 条件ブランチ, 無条件ブランチ等のステートメントを含むことができる。

CMSのコンソールからコマンドがキー・インされるとコマンド・プロセッサはまずEXECを探す。見つからなかった場合には

- 現在トランジェント・エリアにロードされているルーチン
- CMS Nucleus Residentルーチン
- ディスク上のモジュール

の順にルーチンを探す。したがってユーザーはEXECを作成することにより簡単にいわゆるユーザー・コマンドを作成することができる。特にEXEC名がシステムのコマンド名と同じ場合には上記探索順によりEXECの方が実行されるためシステム・コマンド名がキー・インされたときの動作をユーザー特有のものにすることも簡単に行える。

またファイル名PROFILEをもつEXECは特別のEXECとして扱われユーザーがCMSにログオンしたとき(厳密にはログオン時CMSが自動的にIPLされたとき)自動的に実行される。これによりユーザー特有のセット・アップ(例えばファンクションキーに対するコマンドのセット, 必要ならミニ・ディスクのアクセス等)を行うことができる。

EXEC2の簡単な例を図4に示す。更にEXEC2はCMSの提供するエディターであるXEDITのマクロとして作動する能力をもっている。この場合XEDITのマクロはEXEC2のステートメント, XEDITのサブコマンド, CP/CMSコマンドを含むことができる。従ってXEDIT

にユーザー・サブコマンドを追加すること
も容易である。XEDITはXEDIT本来の
コマンド体系の他に従来使用されて
いたEDIT, DESと称する2つのエディ
ターそれぞれと互換性のあるコマンド
体系を準備しているがこれはマクロ機
能により実現している。

GRAB EXEC

```

&TRACE
&IF &N = 0 &GOTO -TELL
&IF &N < 2 &GOTO -BAD
&IF &N > 3 &GOTO -BAD
&IF &N = 2 &ARGS &1 &2 *
COPYFILE &1 &2 &3 &1 &2 A
&EXIT &RC

-BAD &PRINT INVALID GRAB COMMAND
&EXIT 101

-TELL &PRINT COMMAND IS: GRAB FN FT [MODE]
&PRINT COPIES THE GIVEN FILE TO THE A-DISK,
&PRINT AND PASSES BACK THE RETURN CODE FROM
&PRINT 'COPYFILE'.
&EXIT 100

```

図4 EXEC 2の例: GRAB FN FT
とキー入ると指定されたファイル
をA-ディスクにコピーする。

3. CPの提供する機能

ここではCPの提供する機能のうち
特に(CMSを含む)仮想計算機間の
コミュニケーションをサポートする機
能について説明する。

3.1 SPOOL機能

CPは仮想カードリーダー/パンチ、
仮想プリンターへの入出力をディス
クを由実際のリーダー/パンチ、プリ
ンターと入出力するいわゆるSPOOL
機能を提供するが、これに加えて1
つの仮想計算機の仮想パンチ、プリ
ンターへの出力を他の仮想計算機の
仮想リーダーへの入力とする機能
をもつ、相手仮想計算機の指定は
コマンドにより行うことができる。
これにより1つの実計算機内に
ある複数の仮想計算機間でデータ
の受渡しを行うことができる。
(図5)

更にこの機能を拡張したものと

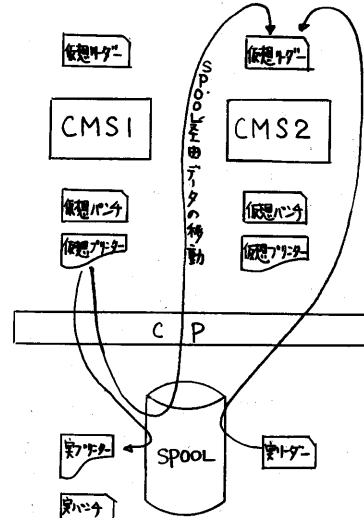


図5 CP SPOOL

RSCSネットワークを使用すると
CP間でのデータ受渡しを行うこと
ができる。この場合には相手先指
定はCPUに対して付けられている名
称とそのCPU内のユーザー名を指
定することにより行う。(図6)

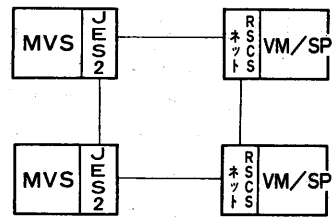


図6 RSCSネットワーク

3.2 VMCF (Virtual Machine Communication Facility)

VMCFはDiagnoseインストラク
ションと仮想外部割込みを使用して2
つの仮想計算機の記憶装置間でデ
ータの送受を行う機能である。カー
ピスの要求はDiagnoseのコードX'68'
により行い相手側仮想計算機は
仮想外部割込みコードX'4001'に
よりいわゆるShoulder Tapping
される。転送されるデータ長に
ついては仮想計算機の仮想記憶
容量を越えなないという以外制限
はな

11.

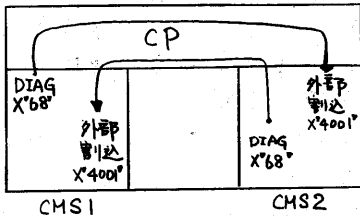


図7 VMCF

両仮想計算機間のデータ転送の方向および受信側の受信確認を必要とするか否かによって3つのプロトコルが定義されている。(図8)

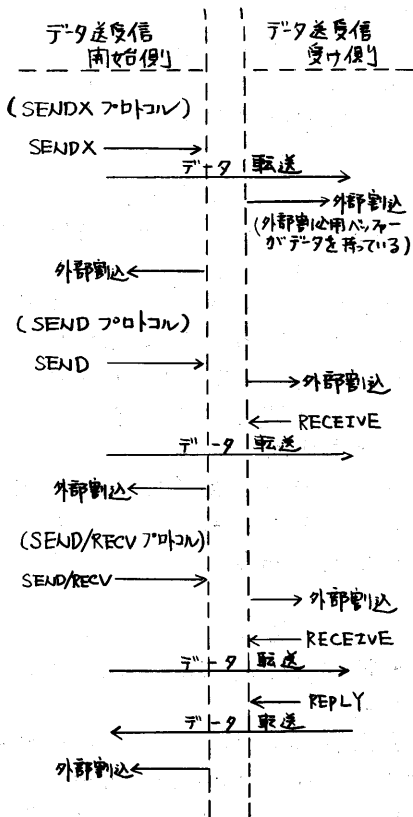


図8 VMCFプロトコル

- SENDXプロトコルは送信データに対する受信確認を必要とする場合
- SENDプロトコルは送信データに対する受信確認を必要とする場合

• SEND/RECVプロトコルは1つのプロトコル内で送受信を行う場合

にそれぞれ使用する。

外部割込みにより受信側が "Shoulder Tapping" されたとき相手仮想計算機名、相手の要求の内容、送信を要求しているデータ長等がコミュニケーション開始に先だて準備されていたバッファにセットされるのでこれにより受信側は要求の内容を知り適当な処理をすることができる。

以上説明したようにVMCFは2つの仮想計算機上で稼動しているプログラム間のコミュニケーション機能を提供するが、VMCF下の拡張として端末ユーザが自分のログオンしている仮想計算機以外の仮想計算機上のプログラムへデータを転送することができるMSG (Special Message) 機能がある。(図9)

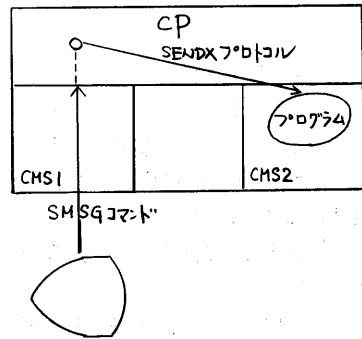


図9 MSG

ユーザがMSGコマンドを受信側仮想計算機名、送信データと共にキーインするとCPがVMCFのSENDXプロトコルを使用してデータを受信側仮想計算機に転送する。

VMCFはCMSによる対話型システムにおいていわゆるサービス仮想計算機を作成するのに有効であり、ISPF (Interactive System Productivity Facility), IFS (Interactiv

e File Sharing), VM/Pass thru等のプログラムで使用されている。

3.3 IUCV (Inter User Communication Vehicle)

IUCVはVMCFと同様に2つの仮想計算機の記憶装置間でデータ転送を行う機能であるがVMCFとはつぎのような相異点がある。

- VMCFではサービスの要求をDiagnoseインストラクションで行うのに対しIUCVではインストラクション・コードX'B2F0'のインストラクションを使用する。X'B2XX'のコードの11ビットはS/370のインストラクション・セットで特権命令に割当てられているがX'B2F0'は定義されていない。従ってVMCFのサービス要求はいわゆるHypervisor CallとしてCPで処理されるのに対しIUCVにおいてはプログラム・チェック・例外の処理として処理される。
- VMCFでは仮想計算機にメッセージ/応答の致着を知らせる方法は外部割込に限られているが、IUCVにおいてはこの非同期的方法に加えて同期的にプログラムからメッセージ/応答の致着を確かめる機能を提供している。
- VMCFではSMSの機能を除いてコミュニケーションは仮想計算機間のみにおいて行われるが、IUCVにおいてはこれに加えてCPのコンポーネントと仮想計算機間のコミュニケーションも可能である。
- 仮想外部割込コードとしてVMCFはX'4001'、IUCVはX'4000'を使用する。VMCFではこの割込コード全体を割込可能/不能とするのみが可能であるが、IUCVでは更にIUCVのサービス要求に対応したサブ機能毎に割込可能/不能

のコントロールを行うことができる。

- IUCVはコミュニケーション・リンクに対しPATHという概念をもち2つの仮想計算機間で複数のPATHを同時に使用することができる。
- VMCFにはこのような概念はない。現在IUCVはSNA (System Network Architecture) コンソール・サポートにおいてCPのコンポーネントであるSNA CCS (Console Communication Service) とVTAM サービス仮想計算機間のコミュニケーションに使用されている。

4. おわりに

VM/370発表以来VMシステム機能拡張は続いており最近ではVM下のMVS仮想計算機のパフォーマンス向上を目指したVM/HPオリリース2, VMに関するDDP (Distributed Data Processing) のための機能拡張をはかったVM/SPリリース2等が使用可能となっている。ここでは特定のリリースにこだわらずにCMS, CPの機能紹介を行ったが一応VM/SPリリース1を想定している。

最後に参考文献として(1)(2)を上げておく、(1)はVMの特集号でVMに関するトピックが乗せられている、(2)はVM開発の歴史である。

[文 献]

- 1) IBM Systems Journal, Vol. 18, (1979)
- 2) R.J. Creasy: The Origin of the VM/370 Time Sharing System, IBM Journal of R&D, Vol. 25, pp.483~490. (1981)