

開発支援ツール使用手引 - バッチ編 -

(Batch Utility User's Guide)

当手引きは、情報開発の標準バッチ・ユーティリティ
の用法を示したものである。

発効日 : 1999年11月 01日
主管部門 : 情報開発.システム開発センター

経緯

- 1987年3月：技術推進から委託され、'87年度 生産性委員会 - 標準部会が原案を作成し、関連部門（情報開発 全ライン管理者および 情報開発企画）の審議（レビュー）を受けて初版発行。
- 1998年11月：2000年問題への対応に伴い、情報開発標準についても全面的見直しを行い第二版発行。
- 1999年9月：2000年対応の関連でBDTに変わる代替ツールFICS(File Interface Control System)を構築。標準として使用法を追加。
- 1999年11月：SCRMSVの機能変更。使用法の記述に考慮点を追加

1 まえがき

1.1 目的・概要	- 1 -
1.2 対象部門・対象業務	- 1 -
1.3 準拠および関連標準	- 1 -
1.4 部門別役割・責任	- 1 -

2 バッチ・ユーティリティ

2.1 PROGM1,2,3	- 2 -
2.2 RPD8DLD	- 3 -
2.3 SCRMSV1 ~ 4	- 3 -
2.4 MSSSCR1,2	- 4 -
2.5 JPD0VV1 (Multi Dataset Copy Utility)	- 5 -
2.6 DCPVV1	- 9 -
2.7 SYMCHK (SYSIN Automatic Generator)	- 10 -
2.8 SYMCHK2 (MEC用 SYMCHK)	- 15 -
2.9 NEWSORT / SORTD	- 17 -
2.10 DCPSORT	- 18 -
2.11 SA00CL	- 20 -
2.12 SC02UEM	- 21 -
2.13 SC02UEML	- 24 -
2.14 SC02CRD2	- 25 -
2.15 SC02CRDR	- 26 -
2.16 FICS (File Interface Control System)	- 27 -

1 まえがき

1.1 目的・概要

使用頻度が高く、共通に使用され、品質が保証された標準バッチ・ユーティリティの使用法を示し、開発・保守の効率向上を図る。

1.2 対象部門・対象業務

情報開発が作成あるいは保守するJCL等で、該当のバッチ・ユーティリティが使用可能な部分に適用する。

1.3 準拠および関連標準

なし

1.4 部門別役割・責任

(1)情報開発.システム開発センター

標準バッチ・ユーティリティの開発・保守を行う。

変更に応じて標準の改訂を行う。

(2)情報開発.各ソリューション

開発実施作業の効率化のため必要に応じて標準バッチ・ユーティリティを使用する。

2 バッチ・ユーティリティ

2.1 PROGM1,2,3

2.1.1 概要

このユーティリティは、コントロールステートメントを読み、シンボリックパラメーターを変換し、その値をインプットとしてIDCAMSを実行しデータセットのアンカログ・スクラッチを行うものである。

使用上の留意点は次の通り。

- シンボリックパラメーターの指定の仕方は、JCL とほとんど同じ。
- DSName Key Word は DSN でもよい。
- 基本的にSYSINはIDCAMS形式で指定する。ただしIEHPROGMの形式でも可とする。(前のバージョンのPROGMxとの互換性を保つため)
- SYSINの変換の際、UT,VOLの指定は一切考慮しない。カタログされていないデータセットは扱えない。

2.1.2 使用方法

- PROGM2 の使い方 その 1 (IDCAMS形式)

```
// EXEC PROGM2,FUNC=AMS,SYM='TM=01'  
DEL      MA01.DS.A01&TM  
DEL      MA01.DS.B01&TM NOSCR    <--UNCATLOGのみの場合
```

- PROGM2 の使い方 その 2 (IEHPROGM形式)

```
// EXEC PROGM2,UT=3380,VOL1=MA01A,VOL2=MA01B,SYM='TM=01'  
SCRATCH  DSN=MA01.DS.A&TM,VOL=&UT=&VOL1,PURGE  
SCR      DSN=MA01.DS.B&TM,VOL=&UT=VOL2,PURGE  
UNCATLOG DSN=MA01.DS.A&TM  
UNC      DSN=MA01.DS.B&TM
```

2.1.3 コントロール・ステートメント

- シンボリックパラメーターの指定の仕方は JCL とほとんど同じ。ただし、シンボリックバリュー中にクォテーションがあってはならない。

2.2 RPD8DLD

2.2.1 概要

Sequential File の Open Close を行う。

2.2.2 使用方法

```
// PGM=RPD8DLD
//STEPLIB DD DSN=SA00.DS.LMOD,DISP=SHR
//MOUNT01 DD DSN=~,
// DISP=~

//MOUNTnn DD DSN=~,
```

- DISP=OLD の場合 Data Set は Empty となる。
- DISP=(,CATLG) の場合 New Allocation & Close
- なお LRECL の制限はない。

2.3 SCRMSV1 ~ 4

2.3.1 概要

- データセットを削除する。
- 個々のデータセット名を指定せず、VOLUME SERIAL を指定し、指定したボリュームに含まれる、保存期間を過ぎたデータセットを削除、アンカタログする。

2.3.2 JCL

```
// EXEC SCRMSVn,UT=unit,VOL1=volume[,VOL2=~ ,
VOL3=~ ,VOL4=~ ]
```

- unit : 削除するデータセットのあるUNITを指定する
- volume : 削除するデータセットのある VOLUME を指定する
 - SCRMSV1 : 対象ボリュームが 1
 - SCRMSV 2 : 対象ボリュームが 2
 - SCRMSV 3 : 対象ボリュームが 3
 - SCRMSV 4 : 対象ボリュームが 4

2.3.3 考慮点

マルチボリュームにカタログされたデータセットの削除、アンカタログにも対応した。ただし、そのデータセットがカタログされているすべてのボリュームに対してSCRMSVを行わないと、カタログのみ削除されてデータのみ残ってしまう危険があるため、原則的にはマルチボリューム指定のデータセットを含むボリュームに対してSCRMSVを使用することは推奨しない。

2.4 MSSSCR1,2

2.4.1 概要

- データセットを DSNAME のみ指定して、アンカタログ、削除できる。

当ユーティリティは、PROGM1,2と同様にIDCAMSを用いてデータセットのアンカタログ・スクラッチするものである。

留意点もPROGM1,2とほぼ同様である。

- パラメータおよびコントロールカードは基本的にはIDCAMSの形式にて記述すること。(使用例1を参照)
- シンボリック・パラメータは、PROGMx と同様に指定できる。
- SYSINのコントロールカードは、前のバージョンのMSSSCR1,2と同様の形式でも可とする。(互換性を保つため。使用例2を参照)

2.4.2 JCL

•MSSSCR2 使用例 1 (IDCAMS形式)

```
//          EXEC  MSSSCR2,FUNC=AMS[,SYM='RUNNO=&RUNNO']
//SYSIN     DD   *
            DEL SP40.RP. ~ [&RUNNO]
            DEL SP40.RP. ~
/*
```

❖ DEL dsname で削除するレポート・データセット名を指定する。

•MSSSCR2 使用例2

```
//          EXEC  MSSSCR2[,SYM='TM=&TM']
//SYSIN     DD   *
            DSN=SM10.DS.XX&TM
            DSN=SM10.DS.YY&TM
/*
```

❖ DSN=dsname で削除するデータセット名を指定する。

2.4.3 制限

VOLUME SERIAL の指定ができないので、カタログされているデータセットにのみ使用できる。

2.5 JPD0VV1 (Multi Dataset Copy Utility)

2.5.1 概要

JCL の指定により、1 ステップで複数のデータセットのコピー (Back up, Load, Unload など) を行うユーティリティである。

また、このユーティリティをマルチタスクで動かすことも可能であり、I/O 装置の I/O バランスを適切に考慮すると、パフォーマンスの良いコピーを行うことができる。

2.5.2 特徴

•処理可能な編成

入力/出力ともに下記が扱える。

- ❖ Physical Sequential (PS)
- ❖ Indexed Sequential (ISAM)
- ❖ VSAM - Keyed Sequential (VSAM - KSDS)
- ❖ VSAM - Entry Sequential (VSAM - ESDS)

ただし、Fixed Length Record に限る。

入力と出力データセットの組み合わせは自由である。

•I/O 装置の種類

テープ装置、ディスク装置を I/O の装置として使用できる。

•出力データセットが ISAM または VSAM - KSDS の場合は、Key の Sequence Check が行われる。

Sequence Error のレコードは、20 レコードまでは無視され、20 レコード以上ある場合、該当データセットの処理は中断される。

Duplicate Check も同様に行われる。

•ユーティリティの終了時に、リターンコードがコンソール上に表示される。

なお、コンソールへの表示を行わないことも可能である。

•EXEC Card のパラメーターの指定により、データセットのマージができる。

2.5.3 JCL

このユーティリティは通常の OS/VS Job として実行する。下記の EXEC ステートメントと DD ステートメントが入出力を指定するために必要である。

EXEC

当ステートメントは次の形式とする。

PGM=JPD0VV1[,PARM= . . .]

パラメーターの指定については次章参照。

STEPLIB DD

当ユーティリティの入っているモジュールライブラリーを指定する。

DSN=SA00.DS.LMOD,DISP=SHR

SYSOUT DD

メッセージデータセットを指定する。

- 入出力データセットのレコード数
- コピーに要した時間
- エラーメッセージ
- シーケンスエラーが起きた場合に、前後のレコードの内容がプリントされる。

XXXXXO DD

入力データセットを指定するステートメント

XXXXXN DD

出力データセットを指定するステートメント

xxxxxO から xxxxxN へ Copy が行われる。

1 ステップにこの組を複数作れば、その数だけコピーが行われる。

CONSOLE DD

ステップへのリターンコード表示を行わない時、DUMMY と指定する。

2.5.4 Parameter

EXEC ステートメントに指定するもので、次の機能がある。指定はオプションである。

- メッセージデータセットのヘディングタイトルを指定
- マージを行う入力データセットの DD 名
- 入出力 DD 名の Suffix を指定

- 記入形式は次のとおり。

PARM='s'

's=AAAA'

's=i'

's=i,AAAA'

's=i,j'

's=i,j,AAAA'

または 'AAAA'

または 'i,AAAA'

または 'i,j'

または 'i,j,AAAA'

開発支援ツール使用の手引 - バッチ編 -

's=i-k'

または 'i-k'

's=i,j-k'

または 'i,j-k'

's=i,j-k,AAAA'

または 'i,j-k,AAAA'

❖ s= サブタスク番号

s が 5 以上の場合：入力ファイルが 0 レコードの場合出力ファイルをオープンしない。

❖ AAAA=Heading Title

最大 80 桁

❖ i,j-k= マージの入力 DD 名 Suffix No.

'i,j' : xxxxxOi と xxxxxOj を入力ファイルとしてマージする。

'j-k' : xxxxxOj から xxxxxOk までを入力ファイルとしてマージする。
(By 1 or By - 1)

•サブタスク番号とマージの入力 DD 名 Suffix No. を指定した場合の、入出力 DD 名の対応は次のようになる。

❖ 指定しない場合

xxxxxO DD・・・Input

xxxxxN DD・・・Output

❖ Suffix i, j

xxxxxOi DD・・・Input (Suffix i)

xxxxxOj DD・・・Input (Suffix j)

xxxxxN DD・・・Output

❖ サブタスク番号 s

xxxxxOs DD・・・Input (サブタスク s)

xxxxxNs DD・・・Output (サブタスク s)

SYSOUTs DD・・・Message Data Set (SYSOUT の代わり)

❖ Suffix i, j, k and サブタスク番号 S

xxxxxOsi DD・・・Input (サブタスク s & Suffix i)

xxxxxOsj DD・・・Input (サブタスク s & Suffix j)

xxxxxOsk DD・・・Input (サブタスク s & Suffix k)

xxxxxNs DD・・・Output (サブタスク s)

SYSOUTs DD・・・Message Data Set (SYSOUTの代わり)

2.5.5 エラーメッセージ

Error Message	Return Code	Process
LRECL NOT EQUAL	10	C
RKP NOT EQUAL	11	C
KEYLEN NOT EQUAL	12	C
DUPLICATE RECORD FOUND	20	C
SEQUENCE ERROR	21	C
UNCORRECTABLE I/O ERROR	50	S
INSUFFICIENT PRIME AREA	55	S
SYNAD ERROR (DCBEXCD=XX)	59	S
TOO MANY ERROR RECORDS FOUND	60	S
LRECL=0	70	S
RKP=0	71	S
KEYLEN=0	72	S
DD CARD DUPLICATE	80	S
DD CARD MISSING	82	S
DATA SET NOT FOUND (DSCB-X ERROR-XX)	85	S

[注] Process C・・・Process Continue

S・・・Error のあった Data Set の組については、Process を Stop する。残りの Data Set の組は処理する。

2.5.6 使用例

•Backup Step

2 つの Data Set のバックアップを 1 本のテープに作る。

```
//BACKUP EXEC PGM=JPD0VV1
//STEPLIB DD DSN=SA00.DS.LMOD,DISP=SHR
//SYSOUT DD SYSOUT=A
//SDXX00 DD DSN=A,DISP=SHR
//SDXX0N DD DSN=A.BACKUP,DISP=(,CATLG),
// UNIT=3400-5,LABEL=RETPD=XX,DCB= ~
//SDYY00 DD DSN=B,DISP=(,SHR),
//SDYY0N DD DSN=B.BACKUP,DISP=(,CATLG),
// VOL=REF=*.SDXX0N,
// LABEL=(2,SL,RETPD=XX),DCB= ~
```

[注] VOL=(,RETAIN) を入れなくても Tape は Rewind しない。

データセット A と B は、PS, IS, VSAM いずれでも良い。

[説明]

❖ データセット A をテープにコピーする (DSN=A.BACKUP)

- ❖ データセット B を上記テープの 2 番目の File としてコピーする
(DSN=B.BACKUP)
- ❖ コピーは出力 DD ステートメントの順番に行われる。

•ISAM / VSAM - KSDS ファイルのロード

```
//LOAD EXEC PGM=JPD0VV1
//STEPLIB DD DSN=SA00.DS.LMOD,DISP=SHR
//SYSOUT DD SYSOUT=A
//SDXX0O DD DSN=BACKUP,DISP=SHR
//SDXX0N DD DSN=ISAM1,DISP=OLD
//SDYY0O DD DSN=VSAMBACK,DISP=SHR
//SDYY0N DD DSN=VSAM,DISP=OLD
//SDZZ0O DD DSN=BACKUP,DISP=SHR
//SDZZ0N DD DSN=ISAM2,DISP=(,CATLG),UNIT=~,
// VOL=SER=~,SPACE=,~,
// DCB=(DSORG=IS,~
```

[説明]

- ❖ データセット BACKUP をすでに割り当てている。
データセット ISAM1 へ ISAM ロードする。
- ❖ データセット VSAMBACK を VSAM へロードする。
VSAM は 前の Step で Define されているか、Reuse Attribute を持つこと。
- ❖ データセット BACKUP を新しいデータセット ISAM2 へ ISAM ロードする
DCB 中に DSORG=IS を書くこと。

2.6 DCPVV1

2.6.1 概要

JPD0VV1 と全く同じ機能を持ち、入出力にDCPの圧縮ファイルを使用できる。

2.6.2 使用方法

使用方法是 JPD0VV1 と全く同様である。 また、DCPファイルのMergeが可能。

[例] 2つのファイルを Merge して、1つの圧縮ファイルに書き出す例。

```
//COPY EXEC DCPVV1,PARM='2-1'
//AAAAO1 DD DSN=PPPI.DS.DATA1,DISP=SHR
//AAAAO2 DD DSN=PPPI.DS.DATA2,DISP=SHR
```

```
//AAAAAN DD DSN=PPPI.DS.NEW,DISP=(,CATLG),MGMTCLAS=TEST,  
// DCB=(RECFM=VBS,LRECL=106,BLKSIZE=6000)
```

2.7 SYMCHK (SYSIN Automatic Generator)

2.7.1 概要

Application により、AS OF DATE, OPERATION DATE CONTROL CARD
などを Data として SYSIN に使用するものがある。

SYSIN Automatic Generator は、これらの Data を Program により自動的に生成
しようとする目的で開発されたものである。

この Program は、該当する Data を Job Card 上に指定された Symbolic
Parameter により作成し、Data は Work 上に書かれる。

この Program を使用するためには、SYSIN のある JOB の頭に 1 STEP を追加す
る必要がある。

2.7.2 JCL

```
//DPCS EXEC SYMCHK,SYM='Parm.List'  
//SYSIN DD *  
Utility Control Statement  
/*  
//STEPN EXEC ~  
//SYSIN DD DSN=&&SYSINX,DISP=(OLD,PASS)
```

❖ 最初の 4 Step は SYSIN Automatic Generator のために追加される STEP
Parm.List

Utility Control Statement で使用する Symbolic Parm を指定する。

Utility Control Statement

Col.1 より Data の Format にあわせ、すべて Symbolic で表示する。

&&SYSINX

Generate された結果が渡される DSN

•使用例 1

```
//DPCS EXEC SYMCHK,SYM='P1=1,P2=2,P3=3'  
//SYSIN DD *  
&P1&P2&P3
```

```
//*
```

```
//STEP EXEC ~
```

```
//SYSIN DD DSN=&&SYSINX,DISP=(OLD,PASS)
```

結果として SYSIN に渡される Data は

```
&&SYSINX      1  2  3
```

```
Col.          1  2  3
```

SYMCHK の STEP は、Control Data を必要とする JOB につき Job 単位に、その JOB の First Step で指定する。

•使用例 2

```
//SP30DPCS JOB (.....),.....
```

```
//DPCS EXEC SYMCHK,SYM='.....'
```

```
//SYSIN DD *
```

```
Utility Control Statement
```

```
//*
```

```
//STEP1 EXEC .....
```

```
//SYSIN DD DSN=&&SYSINX,DISP=(OLD,PASS)
```

```
.....
```

```
//STEP2 EXEC .....
```

```
//SYSIN DD DSN=&&SYSINX,DISP=(OLD,PASS)
```

```
.....
```

```
//STEP3 EXEC .....
```

```
//SYSIN DD DSN=&&SYSINX,(DISP=(OLD,PASS)
```

```
.....
```

```
//STEP4 EXEC .....
```

```
//SYSIN DD DSN=&&SYSINX,DISP=(OLD,DELETE)
```

```
.....
```

[注] STEP4 の SYSIN の DISP で指定されているように、JOB 中 Control Data を使用する最終 Step において、Data Set を Delete する。

2.7.3 Utility Control Statementの使用方法

•当 Program の中ですでに使用されている Symbolic 一覧

□ HY : This Year の上 1 ケタ (98年の場合'9')

LHY : Last Year の上 1 ケタ (97年の場合'9')

T.D : 締日 Default 20日 12月の場合のみ 31日

開発支援ツール使用の手引 - バッチ編 -

YY : Operation Date の Year 2 ケタ
 MM : Operation Date の Month 2 ケタ
 DD : Operation Date の Date 2 ケタ
 J : 和暦 (1998年の場合 '10')
 A : ALPHA.Month (TM=07の場合 'JULY')
 LD : 締日の翌日 通常 21日 1月のみ 1日となる。
 # : 西暦から和暦に直すときの CTL.DATA
 #=88 となっており '98 のとき 98 - 88=10 として平成10
 年が求められる。

[注] 上記 Symbolic は、当 Pprogram で使用しているため User's Job 上では使用できない。

•任意の文字を渡すことが可能

'DATE' という文字を渡す場合

```
//DPCS EXEC SYMCHK,SYM='P1=DATE'
//SYSIN DD *
&P1
/*
//STEPN EXEC ~
//SYSIN DD DSN=&&SYSINX,DISP=(OLD,PASS)
```

&&SYSIN には 'DATE' が渡される

•Conversion

❖ Blank Insertion

DATA にブランクがあるときは、EXEC 上の SYS Parm で '*' を与える。

当 Programでは SYMBOLIC の右辺に使用される '*' はすべてブランクとみなす。

'*' 一個につき 1 ブランク bb07 を渡す場合 (07:TM)

[例] SYM='B2=**,TM=&TM'

Utility Control Statement として &B2&TM

❖ Operation Date の渡し方

[例] 070198 (07=MM, 01=DD, 98=YY) を渡す場合

SYM= . . .

MM, DD, YY は PGM で渡されるため SYM Parm では何も指定しなくてよい。

Utility Control Statement として &MM&DD&YY

❖ 和暦の渡し方

- a) Symbolic で指定した 'TY' の和暦を渡すとき。

[例] 10bbb01 (10: 平成10年 bbb: ブランク3ケ 01: TM)を渡す場合

SYM='J,TY=&TY,B3=***,TM=&TM' (B3=*** : 3ブランク)

Utility Control Statement として &J&B3&TM

- b) Operation Year の和暦を渡すとき。

[例] 10bbbb12 (10: 平成10年 bbbb: ブランク4ケ 12: Operation Month) を渡す場合

SYM='B4=****' (B4=**** : 4ブランク)

Utility Control Statement として &J&B4&MM

[注] SYM Parmに 'J' を指定せずに Utility Control Statement のみに '&J' を指定すれば Operation Year が渡される。

&MM は Default で Operation Month が渡される。

❖ Alph-Month の渡し方

- a) Symbolic Month (TM) の Alph を渡すとき

[例] JULY (TM の Alph) を渡す場合

SYM='A,TM=&TM'

Utility Control Statement として &A

- b) Operation Date の Alph を渡すとき

[例] JUNE (Operation MonthのAlph) を渡す場合

SYM= . . .

(Operation Month は 'MM' として PGM により渡される為不要である)

Utility Control Statement として &A

すなわち、SYM Parm. で 'A' を指定せずに &A を使用したときは、Operation Date の Alph が渡される。

2.7.4 制限

•SYMBOLIC として使用可能個数：16 個

User's Job 内では SYM='~' の中で 16 個まで使用できる。

当 Program 中で下記の Symbolic を使用しているため、User はこれらのものを指定できない。

指定できないもの：HY, LHY, TD, YY, MM, DD, J, A, #, LD (4 . 6 . 3 参照)

•SYM Parm の Byte 数：80 Byte (SYM='~' の中は 80 Byte まで使用可能)

•Utility Control Statement 内の Byte 数：71 Bytes

- Utility Control Statement 内に使用可能 Symbolic 数 : 30個 (種類ではない)
'&TM' が 3 回使用されていれば 3 個となる。

[例] 10 個の例

個数	1	2	3	4	5	6	7	8	9	10
Symbolic	&TM	&SR	&TD	&SR	&HY	&TY	&BK	&BK	&BK	&TD

2.7.5 使用例

- 使用例 1 : 次の Data を渡す場合

番号	1	2	3	4	5	6	7	8	9	10	11	12	13
Data	..	D1	07	01	98	..	JULY	.	01	.	1998	..	10

Col.1

Data の内容 (上記表の番号と対応)

- 1 : 2 ブランク
- 2 : T/C 'D1'
- 3 : Operation Month
- 4 : Operation Date
- 5 : Operation Year
- 6 : 2 ブランク
- 7 : Operation Month の ALPH
- 8 : 1 ブランク
- 9 : Operation Date
- 1 0 : 1 ブランク
- 1 1 : Operation Year の西暦
- 1 2 : 2 ブランク
- 1 3 : Operation Year の和暦

```
//DPCS EXEC SYMCHK,SYM='B2=**,D=D1,B1=*,Y1=1 9'
```

```
//SYSIN DD *
```

```
&B2&D&MM&DD&YY&B2&A&B1&Y1&YY&B2&J
```

```
//*
```

```
//STEPN EXEC . . . . .
```

```
//SYSIN DD DSN=&&SYSINX,DISP=(OLD,PASS)
```

開発支援ツール使用の手引 - バッチ編 -

&MM, &DD, &YY, &A, &Jについては SYSCHK PGM により自動的に渡される

•使用例 2 : 次の Data を渡す場合

番号	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Data	..	D1	07	20	98	..	JULY	.	20	1998	..	10	07	20

Col.1

Data の内容 (上記表の番号と対応)

- 1 : 2 ブランク
- 2 : T/C 'D1'
- 3 : Symbolic で指定した This Month
- 4 : AS OF DATE
- 5 : Symbolic で指定した This Year
- 6 : 2 ブランク
- 7 : Symbolic で指定した This Month の ALPH
- 8 : 1 ブランク
- 9 : AS OF DATE
- 10 : Symbolic で指定した This Year の西暦
- 11 : 2 ブランク
- 12 : Symbolic で指定した This Year の和暦
- 13 : Symbolic で指定した This Month
- 14 : AS OF DATE

```
//DPCS EXEC SYMCHK,SYM='B2=**,D=D1,B1=*,TM=&TM,TY=&TY,A,
B1=*,Y1=19,J'
//SYSIN DD *
&B2&D&TM&TD&HY&B2&A&B2&A&B1&TD&Y1&HY&TY&B2&J&TM
&TD
//*
//STEPN EXEC .....
//SYSIN DD DSN=&&SYSINX,DISP=(OLD,PASS)
```

2.8 SYMCHK2 (MEC用 SYMCHK)

2.8.1 概要

IS Utility SYMCHK の MEC 版が「SYMCHK2」です。

SYSIN 内の &TD について、SYMCHK との相違点は下記の通りです。

ただし、SYMBOLIC として TY, TM、SYSIN として &HY, &TY, &TM, &TD が必要。

•SYMCHK

1 月 ~ 1 1 月 : 2 0

1 2 月 : 3 1 に置き換える。

•SYMCHK2

1 月 : 3 1

2 月 : 2 8 or 2 9

3 月 : 3 1

4 月 : 3 0

5 月 : 3 1

6 月 : 3 0

7 月 : 3 1

8 月 : 3 1

9 月 : 3 0

1 0 月 : 3 1

1 1 月 : 3 0

1 2 月 : 3 1 に置き換える。

当 Utility は、MEC (Month End Closing) プロジェクトの一環として作成したものです。MEC により、日付が月末となるプロジェクトは、当 Utility を使用して下さい。

(日付変更を伴わないプロジェクト、&TD を使わない SYMCHK は今後も使用可能です。)

変更作業は、UPROC 名を SYMCHK から SYMCHK2 に変えるのみで、

SYMBOLIC パラメーターおよび SYSIN DATA に関する変更はありません。

なお、SYMCHK2 を使用するに際して、SYMBOLIC および SYSIN DATA の整合性を CHECK する TSO COMMAND を用意しましたので、ご利用下さい。

2.8.2 使用例

目的 : SYMCHK2 を使用する際の、EXEC PARAMETER および SYSIN DATA を TSO CLIST で SIMULATION し、結果 (BATCH では、&&SYSINX) を確認する。

•SYSIN DATA を直接 KEY - IN する使用例

- - - - TSO COMMAND PROCESSOR - - - -

ENTER TSO COMMAND OR CLIST BELOW:

==>SYMCK2

開発支援ツール使用の手引 - バッチ編 -

```

* SYMCHK2 SIMULATION (FOR MEC USE) *
* 'TD' WILL BE CHANGED TO END OF MONTH *
*****
DEFAULT SYM = #=25,TD=20,LHY=8,HY=8
ENTER ADDITIONAL SYM(AS 'TY=5,TM=11')
      'TY=6,TM=02' (DEFAULT 以外を ' ' でくくって入れる。)
ENTER SYSIN DATA SET NAME OR *
      * (直接 KEY IN するため * を入れる。)
ENTER SYSIN DATA (ENTER /* AFTER SYSIN)
&HY&TY&TM&TD
/*      (必ず /* を ENTER)
860228      結果が DISPLAY される。

```

•SYSIN DATA を DATA Set から READ する使用例

```

- - - - TSO COMMAND PROCESSOR - - - -
ENTER TSO COMMAND OR CLIST BELOW:
==>SYMCK2

*****
* SYMCHK2 SIMULATION (FOR MEC USE) *
* 'TD' WILL BE CHANGED TO END OF MONTH *
*****

DEFAULT SYM = #=25,TD=20,LHY=8,HY=8
ENTER ADDITIONAL SYM(AS 'TY=5,TM=11')
      'TY=6,TM=02'
ENTER SYSIN DATA SET NAME OR *
      "'TSO.UCTRL(SYMCHK)'" ( "' ' で囲む。
                          事前に作成した Data Set / MBR を指定)
860228      結果がDISPLAYされる。

```

2.9 NEWSORT / SORTD

2.9.1 概要

データセットの SORT / MERGE を行うカタブロ。

NEWSORT : SORT

SORTD : MERGE

2.9.2 コントロール・ステートメント

[SORT or MERGE] [FIELDS=(P1,M1,F1,S1,...Pn,Mn,Fn,Sn)
or [FIELDS=(P1,M1,F1,...Pn,Mn,Fn),FORMAT=xx]

- [注] 1 . Col. 2 から始める。
2 . Col. 72 を越える場合は適当な Comma のところで打ち切り、Col. 72 を Non-Blank Character にして次の行の Col.16 より始める。

Pi

Control Field の Top Position

Mi

Control Field の Length

Fi

Control Field の Data Format

CH : Character (EBCDIC)
ZD : Zoned Decimal
PD : Packed Decimal
BI : Binary

Si

Control Field の Sort Sequence

A : Ascending
D : Descending

xx

全 Control Card に共通な Data Format (Fi の項と同じ)

2.9.3 使用方法

•SORT

```
//SORT          EXEC  NEWSORT
//SORTIN        DD   ~
//SORTOUT       DD   ~
//SYSIN         DD   *
sort control card
/*
```

•MERGE

```
//MERGE          EXEC  SORTD
//SORTIN01       DD   ~
//SORTIN02       DD   ~
[/SORTIN03      DD   ~ ]
```

:

```

      :
      [//SORTIN16      DD  ~  ]
      //SORTOUT      DD  ~
      //SYSIN          DD  *
      merge control card
      /*

```

2.10 DCPSORT

2.10.1概要

NEWSORT と全く同じ機能を持ち、JCL DD 上の DCB パラメーターの指定により、SORTOUT の圧縮、非圧縮を行う。

2.10.2使用方法

DATASET 属性 : SORTIN, SORTOUT : 圧縮、非圧縮の混在が可能

•INPUT

1つの場合 : //SORTIN

複数の場合 :

- a. すべてが非圧縮ファイル : //SORTIN DD でコンカチする。
- b. 圧縮/非圧縮ファイルの混在 :
//SORTIN01 - //SORTIN16 (MAX16)で行う。

•OUTPUT

//SORTOUT を用いる。

圧縮 / 非圧縮は、DCB PARAMETERの指定による。

圧縮のとき、

RECFM=VBS, LRECL=実長+6, BLKSIZE=unitの最適ブロックサイズを指定

非圧縮のとき、

RECFM=FB, LRECL=実長, BLKSIZE=実長の整数倍

2.10.3コントロール・ステートメント

CONTROL CARD : カラム 2-71

•OPTIONS STATEMENT

SORTIN CARD の数を指定

DEFAULT は、SORTINS=1(SORT), WAYS=2(MERGE)

SORTIN01-SORTINnn のとき、OPTION SORTINS=nn (または WAYS=nn)

•RECORD STATEMENT

全てに必須

RECORD TYPE=F,LENGTH=(NN).....実レコード長を記入

•SORT / MERGE FIELD

SORT / MERGE STATEMENTは、NEWSORT と同様

SORT : FIELDS=(P1,M1,F1,S1,.....,P64,M64,F64,S64),FORMAT=XX

MERGE : FIELDS=(P1,M1,F1,S1,.....,P64,M64,F64,S64),FORMAT=XX

MERGE FIELDS=(P1,M1,S1,.....,P64,M64,S64),FORMAT=XX

P1

1st CONTROL FIELD の TOP POSITION

M1

1st CONTROL FIELD の LENGTH

F1

1st CONTROL FIELDのDATA FORMAT

CH=CHARACTER (EBCDIC)

ZD=ZONED DECIMAL

PD=PACKED DECIMAL

BI=BINARY

S1

1st CONTROL FIELD の SORT SEQUENCE

A : ASCENDING

D : DESCENDING

XX

全 CONTROL FIELD に共通な DATA FORMAT

[注] カラム 71を越える場合は、適当なカンマのところで打ち切り、カラム 72
に NON-BLANK CHARACTER をパンチし、次のカードのカラム 16より
続ける。

2.11SA00CL

2.11.1概要

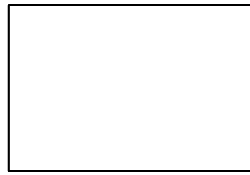
INCB マクロを使用している PL/I プログラムをコンパイル・リンクするサンプ
ル・カタプロである。

コンパイラは Optimizing Compiler である。

2.11.2I/Oサマリー

//EXPD.IN

ソース・プログラム



コンパイル・リスト リンク・リスト

//SYSLMOD

ロード・モジュール・ライブラリー

2.11.3JCL

```
//STEP EXEC SA00CL,  
//          PARM.PLI='xxx,・・・',  
//          PARM.LKED='xxx,・・・',  
//          LMOD='xxx・・・x',  
//          MBR='xxx・・・x'  
//EXPD.IN DD ~
```

PARM.PLI

Default は、'NIS,A,X,AG,M,MAR(2,72,1)'

PARM.LKED

Default は、'XREF,LET,LIST,MAP'

LMOD

ロード・モジュール・ライブラリー

Default は、'TSO.ULMOD'

MBR

ロード・モジュール・ライブラリーに、リンクされるロード・モジュール名。

Default は、'TEMPNAME'

EXPD.IN DD ~

ソース・プログラムのデータセットネーム

2.11.4使用例


```
//STEP    EXEC  SA00CL,
//
//          LMOD='ZA00I.TDS.LMOD',
//          MBR=ZA00A10
//EXPD.IN  DD   DSN=ZA00I.TDS.PGMLIB(ZA00A10),DISP=SHR
```

2.12SC02UEM

2.12.1概要

UEM (Unique Element Master) にエレメントを追加、更新、削除などの登録をするユーティリティである。

2.12.2UEMの特徴

•UEMとは

UEM は情報開発で使用するデータ・エレメントを明確に定義し、記録、維持、管理するシステムである。

•目的

エレメントに関する基本的な情報を一元的に整理し、必要に応じて、これを情報開発関連部門に提供することを目的とする。

•適用範囲

UEMで対象となるものは、情報開発で開発するシステムで使用する全エレメントである。ただし、International Package を除く。

2.12.3JCL

CONTROL CARD (下表左端の * は必須項目)

•E0 ライン

必須	カラム	内 容
*	1-2	'E0'
*	3	ALTERATION CODE
*	4-8	ELEMENT NAME
	10	COUNTRY(未使用)
	11-13	LINE NO.
	14	AUTO NAME REQUEST CODE
	15	ELEMENT STATUS(未使用)
	16	GENERAL USE FLAG
	17	GARBAGE ELEMENT GLAG
*	18	PLI STANDARD ATTRIBUTE
*	19-21	LENGTH
	22	DECIMAL POINT
*	23-67	DESCRIPTIVE TITLE
*	68-71	PROJECT NO.
	72-78	RECORD ID.
	79-80	SEQUENCE NO.

•E1 ライン

必須	カラム	内 容
*	1-2	'E1'
*	3	ALTERATION CODE
*	4-8	ELEMENT NAME
	10	COUNTRY(未使用)
	11-13	LINE NO.
	14	CARD ATTRIBUTE - CODE ATTR.
	15-16	CARD ATTRIBUTE - LENGTH
	17	CARD DECIMAL POINT
	18	PRINT ATTRIBUTE - CODE ATTR.
	19-21	PRINT ATTRIBUTE - LENGTH
	22	PRINT ATTRIBUTE - DECIMAL POINT
	23-37	EDIT PATTERN
	38-47	USER LANGUAGE NAME(未使用)
	48-57	STANDARD HEADER(未使用)
	58-67	LOCAL/REFER ELEMENT NAME
*	68-71	PROJECT NO.
*	72-78	RECORD ID.
*	79-80	SEQUENCE NO.

•E2 ライン

必須	カラム	内 容
*	1-2	'E2'
*	3	ALTERATION CODE
*	4-8	ELEMENT NAME
	10	COUNTRY(未使用)
	11-13	LINE NO.
*	14	TYPE OF DESCRIPTION
*	15-59	CONTENTS DESCRIPTION
*	68-71	PROJECT NO.
*	72-78	RECORD ID.
*	79-80	SEQUENCE NO.

•E3 ライン

必須	カラム	内 容
*	1-2	'E3'
*	3	ALTERATION CODE
*	4-8	ELEMENT NAME
	10	COUNTRY(未使用)
	11-13	LINE NO.
*	14-19	ELEMENT NAME
	20	COUNTRY CODE
*	21	DATA FORMAT SELECTION CODE
	22-24	ARRAY SIZE
	25-57	カラム 14-24の繰り返し
*	68-71	PROJECT NO.
*	72-78	RECORD ID.
*	79-80	SEQUENCE NO.

[注] E0-E3ラインの順番は問わない。

[参照]「DMSS使用者の手引き」

2.12.4使用例

```
//STEP EXEC SC02UEM
//SYSIN DD *
E0AQB 000A F007 BASIC PARTS QUANTITY IN EACH STOCK POINT LD25J
E1AQB 000 LD25J
E2AQB 000N THE TOTAL QUANTITY WHICH THE PARTS MUST BE LD25J
E2AQB 010N CONTAINED IN EACH STOCK POINT LD25J
E2AQB 020N IT IS A KIND OF INVENTORY CONTROL FIGURE LD25J
/*
```

2.13SC02UEML

2.13.1概要

UEM をエレメント名順、またはプロジェクト番号/レコード ID / SEQUENCE 順に印刷するユーティリティである。

2.13.2JCL

CONTROL CARD (下表左端の * は必須項目)

•EA DOCUMENT - エレメントを選別して、エレメント名順に印刷する場合。

必須	カラム	内 容
*	1-2	'EA'
*	3-6	LINE SELECT(該当LINEに1) カラム 3 : E0 カラム 4 : E1 カラム 5 : E2 カラム 6 : E3
*	7-10	PROJECT NO. 'ALL' : 全エレメントを対象 (センター用であり、各人で使用しない)
	11-17	RECORD ID.
	18-61	カラム 7-17の繰り返し
	62-70	SELECT FLAG(未使用)
	71-74	REQUESTING PROJECT NO.
*	75-80	REQUESTING EMP NO. CHECK DIGIT

[注] EA DOCUMENT は 1 枚のみ可

開発支援ツール使用の手引 - バッチ編 -

- EB DOCUMENT - エlementを選別して、プロジェクト毎に印刷する場合。

必須	カラム	内 容
*	1-2	'EB'
*	3-6	LINE SELECT(該当LINEに1) カラム 3 : E0 カラム 4 : E1 カラム 5 : E2 カラム 6 : E3
*	7-10	PROJECT NO. 'ALL' : 全Elementを対象 (センター用であり、各人で使用しない)
	11-17	RECORD ID.
	18-61	REPORT SEPARATOR IMAGE
	62-70	SELECT FLAG(未使用)
*	71-74	REQUESTING PROJECT NO.
*	75-80	REQUESTING EMP NO. CHECK DIGIT

[注] EB DOCUMENT は 20枚まで可

[注意事項] EA、EB を 1 つの INPUT にして実行することも可能

[参照] 「DMSS使用者の手引」

2.13.3使用例

```
//STEP EXEC SC02UEML
//SYSIN DD *
EA1111SQ23
EB1 1 SQ23
/*
```

2.14SC02CRD2

2.14.1概要

CRD (Common Record Description) Input Sheet ('C'ドキュメント)を PDS 上に作成しておき、メンバー指定により CRDを作成するユーティリティである。

2.14.2CRDの特徴

- CRD とは

CRD は UEM に記録されたElementをグルーピングし、プログラムで必要とするレコード・フォーマット (レイアウト) を作成するシステムである。

- 目的

明確に定義されたElement (UEM) と関連付けることにより、フォーマット作

成の効率を高めると共に、相互関係 (エレメント レコード プログラム) を自動的に把握することを目的とする。

•適用範囲

情報開発で開発するホスト系システムで使用する全レコード・フォーマットに適用する。

2.14.3JCL

CONTROL CARD

カラム	内 容
1-2	'CM'
3-9	CRD名: CRD 作成を行いたいPDSのメンバー名を指定する。 [注] メンバー名とCRD名('C'ドキュメントのカラム 2-8)とは一致させること。

[注意事項] 'CM'複数枚により、複数メンバーの指定も可能

[参照] 「DMSS使用者の手引」

2.14.4使用例

```
//STEP EXEC SC02CRD2,CRD='SYS1.TESTCRD'
//SYSIN DD *
CMJRSQEMP
CMJRSQF01
/*
```

2.15SC02CRDR

2.15.1概要

コントロール・カードに、ユーザーのほしいCRDの範囲を指定することにより、CRDを印刷するユーティリティである。

2.15.2JCL

CONTROL CARD (下表左端の * は必須項目)

必須	カラム	内 容
*	1-2	'CA'
*	3-6	COMMAND CODE 'PRT': PRINT REQUEST 'CGEN': DMP2 CONTROL CARD GEN(未使用) 'RGEN': CRD REGEN REQUEST

開発支援ツール使用の手引 - バッチ編 -

*	7	INDICATOR Col.9-24に何を指定するかを示す 'P' : PROJECT NO. 'K' : KEY OF CRD NAME
	8	PRINT MODE ' ' : UEM上の情報は、E0ラインのみ印刷する 'U' : UEM上の情報は、E0 E2ラインを印刷する
*	9-15	PROJECT NO. 特定プロジェクトのCRDが欲しい場合、プロジェクト番号を指定する。 INDICATORは'P'であること。
*	9-13	CRD NAMEの上4桁、または上5桁 CRD名のPREFIXで範囲を指定する場合、CRD名の上4桁または上5桁を指定する。 INDICATORは'K'であること。
	9-23	FIRST CRD NAME - LAST CRD NAME 選択したい対象がある範囲である場合、下限 (カラム9-15) と上限 (カラム17-23) を指定する。

[注意事項] INPUTは複数可能

[参照] 「DMSS使用者の手引」

2.15.3使用例

```
//STEP EXEC SC02CRDR
//SYSIN DD *
CAPRT PUSQ23
CAPRT K JRGFA
CAPRT R JRGFA01 JRGFW01
/*
```

2.16 FICS(File Interface Control System)

2.16.1概要

データセットの送受信を非同期で行う。

2.16.2機能

BDTの機能と追加で新機能がある。

制御テーブルを介してD/S名を意識せずにファイルを受け渡す

複数のファイルの受け渡しが可能

システム間(AIMS<->SIMS)の受け渡しが可能

重複したインターフェイスのファイルは受け渡ししない
 当日分の単純 RERUNは不可能 (パラメーターの変更により可能)
 シンボリック対応
 VSAMファイルにも対応可能

2.16.3 制限、注意事項

FICSを使用する場合は、以下の制限および注意事項があります。

送信側と受信側でINTERFACE IDを決める必要がある
 システム間でデータセットの受け渡しを行う場合は、以下の注意が必要
 送受信どちらかのシステムでJOBを行う必要がある。
 そのため、JOBの分割、OPCの変更等が発生します。

2.16.4 使用方法

送信側 JCL

カタプロFICSENDを使用し、パラメータを指定する。(パラメータの説明は後述)

```
//ST010 EXEC FICSEND,
//  INTID=SA00A001,          <-- INTERFACE ID
//  DSUID=&RUNNO,             <-- DATASET UNIQUE ID
//  DSN='SA00I.TDS.TEST&TD', <-- SEND DATASET NAME
//  SENDER='SA00S01'          <-- SENDER JOB NAME
```

受信側 JCL

カタプロFICSENDを使用し、パラメータを指定する。(パラメータの説明は後述)

受信するデータセットがないための、ダミーアロケーションのステップを追加する。RERUNを行うためには、RERUN=Yと修正が必要です。

RERUN時に受信されるファイルは、RERUNIDがセットされたファイルだけです。

```
//ST010 EXEC FICSRECV,
//  INTID=SA00B001,          <-- INTERFACE ID
//  RERUNID=&TM&TD,           <-- RERUN ID
//  RERUN=N,                  <-- RERUN (Y OR N)
//  DSN='SQ23I.TDS.D&TM&TD', <-- RECEIVE D/S NAME
//  SPTYPE=CYL,               <-- SPACE TYPE
//  SPACE='(3,1)',            <-- SPACE
//  UNIT=3390,                 <-- UNIT
//  VOLUME=SMS001,            <-- VOLUME
//  RECEIVER='SQ23R01'        <-- RECEIVER JOB
//* -----
//DMYALC PROC
//DMYALC EXEC PGM=IEFBR14
//DDDD01 DD DSN=SQ23I.TDS.D&TM&TD,
```

開発支援ツール使用の手引 - バッチ編 -

```
//      DISP=(,CATLG,DELETE),
//      SPACE=(TRK,(1,1),RLSE),
//      DCB=(RECFM=FB,LRECL=80,BLKSIZE=27920)
//      PEND
//  IF  (ST010.RC2.RC = 4) THEN
//ST011  EXEC DMYALC,TM=&TM,TD=&TD
//  ENDIF
```

上記のダミーアロケーションのステップは参考のもの。他の方法でもよい。

ただし実行する場合は、必ず受信のステップのRC=4の場合に実行するようにする。

STATUS確認用 J C L

INTIDのパラメーターをセットする。RERUNIDの有無で受信確認が出来る。

```
/*JOBPARM S=ANY,P=PROC01
//ST010 EXEC FICSSEL,
//      INTID='FICSFIN'      <-- INTERFACE ID
//
```

RECORD削除用 J C L

FICSENDのRERUN時等、特定のRECORDを削除するために使用する。パラメーターは、INTIDとDSUIDのみなので、複数のRECORDを削除することは、出来ない。

```
//ST010 EXEC FICSDEL,      *** ONE RECORD DELETE ***
//      INTID=FICSFIN,      <-- INTERFACE ID
//      DSUID='&TM&TD'      <-- DATASET UNIQUE ID
```

2.16.5 パラメータ

送信側

パラメーター名	属性	説明
INTID	CHAR(8) 上 4 桁受信側Appli#下 4 桁英数字	送受信INTERFACE 受信側と同じにする。
DSUID	CHAR(5) 英数字 2 桁 + RUNNOなど	INTIDと共にD/SをUNIQUEにする。英数字 2 桁は、RUNNOでUNIQUEにならないとき使用
DSN	CHAR(50)	受信側に渡すD/S名
SENDER	CHAR(8)	送信用STEPをコーディングしているJOB名

受信側

パラメーター名	属性	説明
INTID	CHAR(8)	送受信INTERFACE 送信側と同じにする。

開発支援ツール使用の手引 - バッチ編 -

	上 4 桁受信側Appli#下 4 桁英 数字	
RERUNID	CHAR(5) RUNNOなど	空か、当日分のとき受信する。
DSN	CHAR(50)	受信するD/S名
RECEIVER	CHAR(8)	受信用STEPをコーディングしているJOB名
SPTYPE	CHAR(4)	受信するD/SのSPACE TYPE CYL --->シリンダー TRA --->トラック
SPACE	CHAR(9)	受信するD/SのSPACE (XX,XX)という形式で記述
VOLUME	CHAR(6)	受信するD/SのVOLUME デフォルトでは、SMS01
UNIT	CHAR(4)	受信するD/SのUNIT デフォルトでは、3390
RERUN	CHAR(1)	RERUN 時には、Yをセット。 日々のRUN時は、Nをセット。