

I – プログラム・モジュール、パネル・モジュール格納データセットの作成

- ① ロード・モジュール格納ライブラリーを作成する。

```
//LOADLIB DD DISP=(NEW,CATLG),DSN=hlq.LLIB,  
//          UNIT=SYSDA,VOL=SER=volser,SPACE=(TRK,(5,,10)),  
//          DCB=(RECFM=U,BLKSIZE=32760)
```

hlq … 第1修飾子。任意の修飾子を指定する。

- ② パネル・モジュール格納ライブラリーを作成する。

```
//PANELIB DD DISP=(NEW,CATLG),DSN=hlq.PLIB,  
//          UNIT=SYSDA,VOL=SER=volser,SPACE=(TRK,(3,,10)),  
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=0)
```

- ③ CLIST 格納ライブラリーを作成する。

```
//CLSTLIB DD DISP=(NEW,CATLG),DSN=hlq.CLIB,  
//          UNIT=SYSDA,VOL=SER=volser,SPACE=(TRK,(2,,10)),  
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=0)
```

- ④ オブジェクト・モジュール格納ライブラリーを作成する。

```
//OBJLIB DD DISP=(NEW,CATLG),DSN=hlq.OBJ,  
//          UNIT=SYSDA,VOL=SER=volser,SPACE=(TRK,(5,,10)),  
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=0)
```

II – 配布モジュールのアップロード

- ① Object フォルダ内のすべてのファイルを、作成した hlq.OBJ にバイナリー転送でアップロードする。

TPM、TPMAUTH、TPMTEST

- ② Panel フォルダ内のすべてのファイルを、作成した hlq.PLIB にバイナリー転送でアップロードする。

TKSPBREK、TKSPHELP、TKSPMAIN、TKSPNOTE

III - ロード・モジュールの作成

次の JCL を参考にして、PM のロード・モジュールをリンケージする。

```
//BINDTPM EXEC PGM=IEWL,  
//          PARM='LIST,MAP,XREF,NCAL,AMODE=31,RMODE=ANY'  
//SYSPRINT DD SYSOUT=*  
//SYSLMOD DD DISP=SHR,DSN=hlq.LLIB  
//SYSOBJ DD DISP=SHR,DSN=hlq.OBJ  
//SYSLIN DD *  
  INCLUDE SYSOBJ(TPM)  
  ENTRY TPMMAIN  
  NAME PM(R)  
  INCLUDE SYSOBJ(TPMAUTH)  
  SETCODE AC(1)  
  NAME TPMAUTH(R)  
  INCLUDE SYSOBJ(TPMTEST)  
  MODE AMODE(24),RMODE(24)  
  NAME TPMTEST(R)  
//
```

バインダー（リンケージ・エディター）のステップは CC=4 で終了する。

IV - ロード・モジュール・ライブラリーの APF 登録

hlq.LLIB を APF 許可ライブラリーとして登録する。

```
MVS コマンドで登録  
=====  
SETPROG APF,ADD,DSN=hlq.LLIB,VOL=volser
```

TPM を継続して使用する場合は、SYS1.PARMLIB の PROGxx メンバーに追加するとよい。

コンソール（SDSF の LOG パネル含む）の操作権限を持たない場合は、システム管理者に相談する。

V - TSO/E の許可プログラム登録

PM 起動時に初期設定処理で実行する TPMAUTH モジュールを、TSO/E 許可プログラムとして登録する。使用中の IKJTSOxx メンバーのサフィックス xx 部分は、IEASYSxx の IKJTSO パラメーターの指定を見るか、「D IKJTSO,AUTHPGM」コマンドを実行すれば確認できる。

```
SYS1.PARMLIB(IKJTSOxx)
=====
AUTHPGM NAMES(      /* AUTHORIZED PROGRAMS      */ +
  TPMAUTH           /* TSO PROGRAM MONITOR */ + ←
  ICHUT100         /*                   */ +
  :
  IEBCOPY          /*                   */ +
  *)
```

この行を追加する。

登録したら、次の MVS コマンドで反映できる。

```
SET IKJTSO=xx
```

反映されたかどうかは、「D IKJTSO,AUTHPGM」コマンドを実行すれば確認できる。

VI - PM パネルで使用するファンクションキー・リストの登録

使用している ISPF プロファイルの中に、PM ツールの表示パネルで使用するファンクションキー・リストを登録する。

```
Log/List Function keys Colors Environ Workstation Identifier Help
ISPF Settings
Command ==>
Options
Enter "/" to select option
Z Command line at bottom
Z Panel display CUA mode
Z Long message in pop-up
Z Tab to action bar choices
Z Tab to point-and-shoot fields
Z Restore TEST/TRACE options
Z Session Manager mode
Z Jump from leader dots
Z Edit PRINTDS Command
Z Always show split line
- Enable EURO sign
Print Graphics
Family printer type 2
Device name . . . .
Aspect ratio . . . 0
General
Input field pad . . B
Command delimiter . ;
Terminal Characteristics
Screen format 3 1. Data 2. Std 3. Max 4. Part
Terminal Type 3 1. 3277 2. 3277A 3. 3278 4. 3278A
5. 3290A 6. 3278T 7. 3278CF 8. 3277KN
```

ISPF のオプション 0 を実行し、アクション・バーの Function Keys のポップアップを開く。(日本語パネルの場合は、機能キーと表示されている)


```

- File View Keylist Utility -
- Keylist Utility -
C Change Keylist 4 of 15
O PAGE +
A The current keylist is for application id ISR
Change to application ID . . TKS

ISRNSAB SHARED
ISRREFL SHARED
ISRREFO SHARED
ISRSAB SHARED *** Currently active keylist ***
ISRSLAPP SHARED
ISRSNAB SHARED
ISRSPBC SHARED
ISRSPEC SHARED
ISRSTRTP SHARED
ISRTASKL SHARED
ISRTEST SHARED
T

```

application ID (適用業務 ID) を「TKS」に変更する。

TKS と上書き入力して実行キーを押すと、空のキーリストが表示される。

```

- File View Keylist Utility -
- Keylist Utility for TKS -
C 1. New... Scroll ==> PAGE +
O 2. Edit...
3. View...
4. Delete... Edit V=View D=Delete /=None
5. Exit
**End**
T

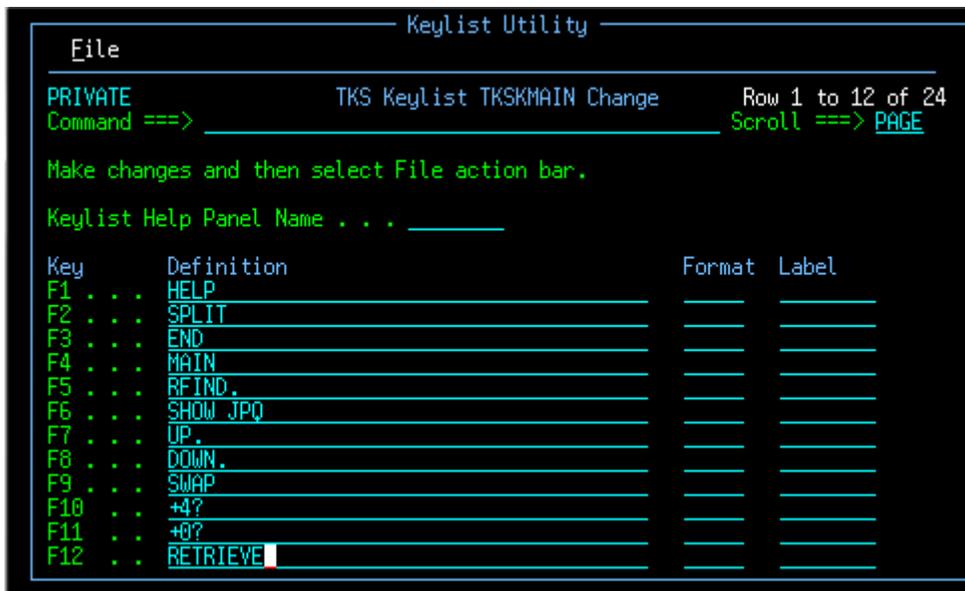
```

アクション・バーの File のポップアップを開く。

ポップアップが開いたら、「1」を選択する。



Keylist Name (キーリスト名) に、「TKSKMAIN」を入力する。



左の画面に従って、同じ内容を設定する。

一部のコマンドは、名前の末尾にピリオドがある。これも忘れずに入力のこと。

F12 まで入力し終わったら、実行キーを押す。

(PM パネルでは画面のスクロールやデータの探索をプログラム独自に行っている。ピリオドを付けずに登録するとコマンドを ISPF が先に拾ってしまうため、区別するためにピリオドを付けて別コマンドとなるようにしている)

```

File                                     Keylist Utility
-----
PRIVATE                                TKS Keylist TSKMAIN Change      Row 1 to 12 of 24
Command ==> _____ Scroll ==> PAGE

Make changes and then select File action bar.

Keylist Help Panel Name . . . _____

Key  Definition                               Format  Label
F1 . . . . . HELP                               LONG   HELP
F2 . . . . . SPLIT                              LONG   SPLIT
F3 . . . . . END                                LONG   END
F4 . . . . . MAIN                               LONG   MAIN
F5 . . . . . RFIND.                             LONG   RFIND.
F6 . . . . . SHOW JPQ                           LONG   SHOW JPQ
F7 . . . . . UP                                  LONG   UP
F8 . . . . . DOWN.                              LONG   DOWN.
F9 . . . . . SWAP                               LONG   SWAP
F10 . . . . . +4?                               LONG   +4?
F11 . . . . . +0?                               LONG   +0?
F12 . . . . . RETRIEVE                          LONG   RETRIEVE

```

登録するとこのような画面になる。

F1 から F12 まで登録できたら、PF3 キーで終了。

キー設定の内容は、好みで自由にカスタマイズできる。具体的なコマンド名は、ヘルプや操作ガイドを参照する。ただし、F1、F2、F3、F7、F8、F9 あたりは ISPF の標準キー設定と異なってしまうため変更しない方がよい。F3 キーや F9 キーを変えてしまうと F3 キーでの終了や F9 キーでの画面切り替えができず、END や SWAP と入力しなければならなくなる。ほとんどのキーをカスタマイズするなら、F1 から F12 ではなく F13 から F24 に追加設定する方がよい。

```

File View                               Keylist Utility
-----
Keylist Utility for TKS                 Keylist saved
Command ==> _____ Scroll ==> PAGE
Actions: N=New E=Edit V=View D=Delete /=New

Keylist  Type
- TSKMAIN PRIVATE
**End**

```

ここに N を入力。

メインのキーリストが登録できたら、次はヘルプ用のキーリストを作る。

TKSKMAIN の左横に、「N」を入力して作業を続ける。



ヘルプ用のキーリスト名は、「TKSKHELP」



ヘルプ用のキーリスト内容。入力したら、実行キーで登録。
登録したら、作業は終わり。PF3 キーで終了していく。

F7 と F8 が UP と DOWN でなく LEFT と RIGHT になっているが誤りではない。
画面の内容通りに定義すること。

```
Keylist Utility
File View
-----
Keylist Utility for TKS          Row 1 to 2 of 2
Command ==>                      Scroll ==> PAGE
Actions: N=New E=Edit V=View D=Delete /=None
Keylist      Type
TKSKHELP    PRIVATE
TKSKMAIN    PRIVATE
*End**

```

ここにNを入力。

ヘルプのキーリストも登録できたら、次はノート用のキーリストを作る。
TKSKHELP の左横に、「N」を入力して作業を続ける。

```
Keylist Utility
File View
-----
Keylist Utility
Keylist Name Prompt
Enter a keylist name and press enter to perform action.
Keylist Name . . . TKSKNOTE

```

ノート・パネル用のキーリスト名は、「TKSKNOTE」

```

Keylist Utility
File Defaults
TKS Keylist TKSNOTE Create Row 1 to 12 of 24
Command ==> Scroll ==> PAGE
Make changes and then select File action bar choice.
Keylist Help Panel Name . . .
Key Definition Format Label
F1 . . . HELP
F2 . . . SPLIT
F3 . . . END
F4 . . .
F5 . . .
F6 . . .
F7 . . . UP
F8 . . . DOWN
F9 . . . SWAP
F10 . . .
F11 . . .
F12 . . . RETRIEVE

```

ノート・パネル用のキーリスト内容。入力したら、実行キーで登録。

こちらは、F7 と F8 を UP と DOWN で定義する。
画面の内容通りに定義すること。

```

Keylist Utility
File Defaults
TKS Keylist TKSNOTE Create Row 1 to 12 of 24
Command ==> Scroll ==> PAGE
Make changes and then select File action bar choice.
Keylist Help Panel Name . . .
Key Definition Format Label
F1 . . . HELP LONG HELP
F2 . . . SPLIT LONG SPLIT
F3 . . . END LONG END
F4 . . .
F5 . . .
F6 . . .
F7 . . . UP LONG UP
F8 . . . DOWN LONG DOWN
F9 . . . SWAP LONG SWAP
F10 . . .
F11 . . .
F12 . . . RETRIEVE LONG RETRIEVE

```

TKSKMAIN、TKSKHELP、TKSKNOTE の3つのキーリストを登録したら、作業は終わり。PF3 キーで終了していく。
キーリストが作成されると、個人用 ISPF プロファイル・データセットの中に、TKSPROF というメンバーが作成されている。複数のユーザーが PM を利用する場合、個々のユーザーでキーリストを作らなくても ISPF プロファイル・データセットのメンバーTKSPROF をコピーしてもよい。ただし、PM の標準コマンド設定以外のコマンドにカスタマイズされていると、その内容でコピーされることになる。

※ISPF 起動プロシーチャーの修正ができない場合（管理者に許可されないなど）は、手順 VIII に進む。

使用している ISPF 起動プロシーチャーを次のように修正する。プロシーチャーの中で、ISPLIB を割り振っている ALLOC コマンドを探す。

```
      :
ERROR OFF
CONTROL NOMSG
TSOLIB RESET
CONTROL MSG
TSOLIB ACT DATASET(' hlq.LLIB' )
ERROR EXIT
ALLOC FI(ISPLIB) SHR DA( +
    ' hlq.PLIB' +
    ' ISP.SISPPENU'      +
    ' ISF.SISFPLIB'     +
    ' ISP.SISPPJPN' )
```

この行を追加する。

※すでに別の TSOLIB コマンドが使われている場合は、その内容と矛盾しないように個別に調整する。

プロシーチャーの修正が終わったら、ISPF を再起動する。次は、手順 X に進む。

VIII - パネル・ライブラリーのセットアップ用 CLIST の作成 (ISPF 起動プロシーチャーが修正できない場合にのみ作業する)

hlq.CLIB にメンバー名 SETUPTPM で、次の内容の CLIST を登録する。

```
ISPEXEC LIBDEF ISPLIB DATASET ID(' hlq.PLIB' )
```

作った CLIST を実行する。オプション 6 のコマンド・シェルより実行。

```
EX ' hlq.CLIB(SETUPTPM)'
```

ISPLIB に追加されたかは、ISPLIBD コマンドで確認できる。

IX - ロード・モジュール・ライブラリーを TSO セッションのリンク・ライブラリーに追加する (ISPF 起動プロシーチャーが修正できない場合にのみ作業する)

通常、ISPF のダイアログ・プログラムは、手順 VIII で使った LIBDEF サービスで、ISPLIB に追加登録すればよいのであるが、PM ツールでは内部の処理で TSO の IKJEFTSR サービスを使用して手順 V で登録した TPMAUTH モジュールを呼び出している。このためには、TPMAUTH モジュールが格納されたライブラリーが、STEPLIB、TSOLIB あるいは MVS のリンク・ライブラリーのいずれかに連結されている必要がある。

① TSO のログオン・プロシーチャーが修正できる場合。

ログオン・プロシーチャー内の STEPLIB に hlq.LLIB を追加定義する。STEPLIB がなければ新規に追加する。

```
// EXEC PGM=IKJEFT01,DYNAMNBR=256
//STEPLIB DD DISP=SHR,DSN=...your system library...
// DD DISP=SHR,DSN=hlq.LLIB
```

この行を追加する。

ログオン・プロシーチャー修正後、TSO に再ログオンする。

② TSO のログオン・プロシージャーは修正できないが、TSO の READY プロンプトが表示できる場合。

(ISPF を自動起動ではなく、READY プロンプトから ISPF 起動プロシージャー名を入力することで ISPF を起動できる場合)

READY プロンプトから、TSOLIB コマンドを使い、追加のロード・モジュール探索経路を設定する。

```
TSOLIB ACT DATASET('hlq.LLIB')
```

TSOLIB コマンド実行後、ISPF を再起動する。

※TSOLIB コマンドは、ISPF のコマンド・シェルからは実行できない。

①か②のいずれかが選択できたなら、次は、手順Xに進む。どちらも選択できなければ手順IXに進む。

IX - ロード・モジュール・ライブラリーの LNKLST 登録 (ISPF 起動プロシージャーが修正できない場合にのみ作業する)

TSO セッションの STEPLIB にも追加できず、かつ TSO にログオン後 ISPF が自動起動される、ISPF 終了後は自動 LOGOFF される、など TSO の READY プロンプトが表示できない場合は、hlq.LLIB を、MVS のリンク・ライブラリーとして MVS の LNKLST に追加することになる。

hlq.LLIB を LNKLST に追加する。(コンソール (SDSF の LOG パネル含む) の操作権限を持たない場合は、システム管理者に相談する)

MVS コマンドで追加

=====

① 現行の LNKLST セット名を確認する。

```
D PROG, LNKLST, JOBNAME=userid
```

② hlq.LLIB を追加する新しい LNKLST セット名を登録する。

```
SETPROG LNKLST, DEFINE, NAME=TMP_LNKLST, COPYFROM=current name
```

③ 登録した新しい LNKLST セットに hlq.LLIB を追加する。

```
SETPROG LNKLST, ADD, NAME=TMP_LNKLST, DSN=hlq.LLIB
```

④ 新しい LNKLST セットをアクティブにして、自分の TSO セッションに反映させる。

```
SETPROG LNKLST, ACTIVATE, NAME=TMP_LNKLST
```

```
SETPROG LNKLST, UPDATE, JOB=userid
```

⑤ 元へ戻す。

```
SETPROG LNKLST, ACTIVATE, NAME=previous name ①で確認した名前を指定
```

hlq.LLIB が追加された LNKLST セットは、TSO からログオフすると無効になる。
再ログオン後に、④を再び行う必要がある。

X - PM (TSO プログラム実行モニター) ツールの起動テスト

PM を起動できる準備が整ったら、動作確認用のプログラムを使って PM ツールを起動する。

ISPF のオプション 6 のコマンド・シェル・パネルで、次のコマンドを実行する。

```
PM TPMTEST
```

実行すると、現行画面に次のメッセージが出る。

```
HELLO WORLD
```

```
*** █
```

そのままエンター・キーを押下する。

```
BREAK MONITOR (R3.13) -----
>>TSO PROGRAM MONITOR<< ALLOC: 000788 KB REAL:00000000001CDC0 RKEY:8FRC HOM
ASCB:00F8CA00 TCB:009BCB28 RB:009BCA58 CDE:009BCEB8 PROGRAM:00042CA8 (TPMTEST)
NR1-4: 00000000-00000000 00000000-00000000 00000000-00000000 00000000-00000000
CMD ==> █

PSW AT TIME OF BREAK: 078D0000 80042DC0 ILC: 02 INTC: 00CA CMPC: -----
GR00: 01195887 GR01: 0111345F GR02: 07830F98 GR03: 0003FF00
GR04: 0003FE98 GR05: 07830F30 GR06: 0003FE30 GR07: 0003FDC8
GR08: 00035400 GR09: 00035000 GR10: 87873FE4 GR11: 056DAD38
GR12: 00042CA8 GR13: 0003FF68 GR14: 80FC8308 GR15: 00000000

00042DC0 VKEY      HEX-DATA      CHR-DATA
00042DC0 (88)    1FFF0A03 41F0007B 07FE4400 C1221D11    ... 0 #..à A...
00042DD0 (88)    411001C8 41000080 89000018 16100A0D    ..H .i .....
00042DE0 (88)    41F0F001 47F0C138 41000001 4110C14C    00.ã0A. .:A<
00042DF0 (88)    0A010000 00000000 010D47F0 C18C0000    .. ..ã0Aã
00042E00 (88)    00000000 00000000 00000000 00000000
00042E10 (88)    00000000 00000000 00000000 00000000
00042E20 (88)    00000000 00000000 00000000 00000000
00042E30 (88)    00000000 4110C158 D7331000 10009640    .AiP... o
00042E40 (88)    10019604 1002D207 1004C2D0 41F00014    ..o...K...B} 0 .
```

プログラムに埋め込まれたブレークポイントによって、パネルが切り替わり、実行が中断している状態。

このパネルで、実行を中断したテスト・プログラムの PSW、レジスター、メモリーの内容を表示できる。F1 キーでヘルプの表示、ヘルプの終了は F3 キー。

CMD 欄に GO と入力するか、F3 キーで終了し、次のメッセージ表示となる。

```
TPM910D TPMTEST ENDED CODE=0
```

```
*** █
```

ここでエンター・キーを押下すれば、コマンド・シェル・パネルに戻る。

(PM の操作方法は、PM オペレーション・ガイドを参照)

プログラムの実行サマリーと実行環境に関する状況メッセージが、TSO セッションの SYSOUT に出力されている。SDSF の ST パネルで、自分の userid 空間を選択して、?文字を NP フィールドに入力すれば SYSnnnnn の DD 名で SYSOUT に出力されているのが確認できる。

PM ツールの起動に必要な定義を、ISPF 起動プロシージャー内に行っていない場合は、TSO にログオンし ISPF を起動する毎に必要なセットアップ操作を繰り返すことになる。継続して PM ツールを使うならば、システム管理者に相談して PM ツールの起動に必要な定義を追加してもらうか、追加された新しい ISPF 起動プロシージャーを用意してもらうことを奨める。