COMMANDE

1) CONTROLE DE VALIDITE D'UN PROGRAMME

Appel d'un CL (VLDCKRC2) qui contrôle la validité du paramètre saisie. Le paramètre à vérifier est DATE.

```
             CMD
             PARM       KWD(DATE) TYPE(*CHAR) LEN(6) DFT(*TODAY) +
                          RANGE('000000' '999999') SPCVAL((*TODAY 0))
```

Le CL VLDCKRDC2 est un CL qui vérifie la validité de la date. Si la date est incorrecte l'exécution de la commande s'arrêtera en affichant le message provenant de VLDCKRDC2.Si la date est correcte l'exécution de la commande continuera en appelant le CL VLDCKRDC1

Pour compiler la commande :

```
                     CRTCMD CMD(VLDCKR) PGM(VLDCKRC1) VLDCKR(VLDCKRDC2)
```

2) ENVOIE DE PARAMETRE A BLANC

On appelle la commande qui appel un cl sans avoir à saisir des paramètres. Ces paramètres seront renseignés dans le cl par une auto-soumission (ex ANCCFR10)

```
             PARM       KWD(SOC) TYPE(*CHAR) LEN(2) CONSTANT('  ')
```

3) RETOUR DE PARAMETRES

RSTD : valeur limitative
RTNVAL : valeur renvoyée par le CPP
SPCVAL : valeur spéciales
MIN : nombre minimal de valeur

```
             PARM       KWD(FROMFMT) TYPE(*CHAR) LEN(10) RTNVAL(*NO) +
                          RSTD(*YES) SPCVAL((*CURRENT) (*JOB) +
                          (*SYSVAL) (*DMY) (*DMYC) (*DMYS) (*MDY) +
                          (*MDYC) (*MDYS) (*YMD) (*YMDC) (*YMDS) +
                          (*JUL) (£COJO) (£NSEM) ($MY) ($YM)) +
                          MIN(1) PROMPT('Format d''origine')
             PARM       KWD(TODATE) TYPE(*CHAR) LEN(16) RTNVAL(*YES) +
                          MIN(1) PROMPT('Variable pour TODATE      +
                          (16)')
```

4)

Lors de la création de commande si dans un paramètre

```
             ==> RTNVAL(*YES)
       CRTCMD    ALLOW(*IPGM *BPGM)
```

Utilisation de UIM aide lors de la création CMD

```
                  CRTCMD PGM(NAME) HLPPNLGRP(NAME) HLPID(*CMD)
```

Utilisation de Programme de Contrôle de validité de la CMD doit contenir les mêmes paramètres que le PGM

```
                  CRTCMD PGM(NAME1) VLDCKR(NAME0)
```

Création de commande personnelle utilisant QCMDEXC

```
                  CRTCMD CMD(WS)  PGM(QCMDEXC)

             CMD      ('WS')
             PARM     KWD(CMD) TYPE(*CHAR) LEN(9) +
                          CONSTANT(WRKSBMJOB)
             PARM     KWD(LEN) TYPE(*DEC) LEN(15 5) CONSTANT(9) MIN(1)
```

--

Emploie de qualificatif de zone pour TYPE

```
             PARM       KWD(FILE) TYPE(LIB_FILE) MIN(1) FILE(*UPD) +
                          PROMPT('Nom du fichier' 1)

  LIB_FILE:  QUAL       TYPE(*NAME) LEN(10) MIN(1) EXPR(*YES)
             QUAL       TYPE(*NAME) LEN(10) DFT(*LIBL) +
                          SPCVAL((*LIBL) (*CURLIB)) MIN(0) +
                          EXPR(*YES) PROMPT('Bibliothéque')

Emploie du contrôle d'invite PMTCTL

  EX1: NBRTRUE si vérifier
             PARM       KWD(CLNOPT) TYPE(*CHAR) LEN(8) RSTD(*YES) +
                          DFT(*REMOVE) VALUES(*ARCHIVE *REMOVE) +
                          PROMPT('option choisie:' 4)
(ne s'affiche que si la condition PMTCTL est remplie) ==>
             PARM       KWD(ARCLIB) TYPE(*NAME) PMTCTL(CLNOPT) +
                          PROMPT('Archive Library:' 5)

  CLNOPT:    PMTCTL     CTL(CLNOPT) COND((*EQ *ARCHIVE)) NBRTRUE(*EQ 1)

             DEP        CTL(&CLNOPT *EQ *ARCHIVE) PARM((ARCLIB))

  EX1: LGLREL si relation logique remplie
             PARM       KWD(P1) TYPE(*CHAR) LEN(5) RSTD(*YES) +
                          VALUE(*ALL *SOME *NONE)
             PARM       KWD(P2) TYPE(*NAME) LEN(10) SPCVAL(*ALL)
             PARM       KWD(P3) TYPE(*CHAR) LEN(10) PMTCTL(LOG1)

L'apparition de P3 ne se fera que si :
          *ALL est spécifier pour P1
          *SOME est spécifier pour P1 et *ALL est spec. pour P2
          *NONE est spécifier pour P1 et *ALL n'est pas spec. pour P2
  LOG1:      PMTCTL     CTL(P1) COND((*EQ *ALL)
             PMTCTL     CTL(P1) COND((*EQ *SOME) LGLREL(*OR)
             PMTCTL     CTL(P2) COND((*EQ *ALL) LGLREL(*AND)
             PMTCTL     CTL(P1) COND((*EQ *NONE) LGLREL(*OR)
             PMTCTL     CTL(P2) COND((*NE *ALL) LGLREL(*AND)

*-------------------------------------------------------------------------------------
Emploie de définition dépendante DEP(avec message), Création de cmd effectué avec Fichier MSG
               CRTCMD MSGF(QUSERMSGF)
 EX1:
             PARM       KWD(LEN) TYPE(*DEC) LEN(2) MIN(1) +
                          RANGE(1 16) +
                          PROMPT('Longueur de la variable')

             PARM       KWD(TYPE) TYPE(*CHAR) LEN(5) DFT(*CHAR) +
                          VALUES(*CHAR *DEC *DATE) RSTD(*YES) +
                          PROMPT('Type')
Concordance entre deux paramètre TYPE et LEN

             DEP        CTL(&TYPE *EQ '*DATE') PARM((&LEN *EQ 6)) +
                          NBRTRUE(*EQ 1) MSGID(USR9999)
 EX2:

             PARM       KWD(LOWRNG) TYPE(*CHAR) LEN(16) DFT(*NONE) +
                          PROMPT('value INF.')
             PARM       KWD(HIRNG) TYPE(*CHAR) LEN(16) DFT(*NONE) +
                          PROMPT('value SUP.')
             PARM       KWD(VALUES) TYPE(*CHAR) LEN(16) +
                          MAX(12) +
                          PROMPT('Liste de Valeurs')
Existence de condition de paramètre (LOWRNG/HIRNG)
             DEP        CTL(VALUES) PARM(LOWRNG HIRNG)  +
                          NBRTRUE(*EQ 0) MSGID(USR9990)

Utilisation de RANGE dans un paramètre numérique
             PARM       KWD(DATE) TYPE(*DEC) LEN(6 0) DFT(*TODAY) +
                          RANGE(000000 999999) SPCVAL((*TODAY 0)) +
                          PROMPT('Date (sys fmt)           (6 0)')
```

```
Utilisation de *PMTRQS => F10 paramètre supplémentaire
              PARM       KWD(RETAIN) TYPE(*CHAR) LEN(4) RSTD(*YES) +
                           DFT(*YES) VALUES(*YES *NO) EXPR(*YES) +
                           PMTCTL(*PMTRQS) PROMPT('Gardé fichier')

Utilisation de Programme de contrôle invite (PGMTEST)
              PARM       KWD(FROMFILE) TYPE(FROM) +
                           PMTCTLPGM(PGMTEST) PROMPT('Fichier d''origine' 1)

  FROM:       QUAL       TYPE(*NAME) LEN(10) DFT(*DFT) +
                           SPCVAL((*DFT) (FILE01))
              QUAL       TYPE(*NAME) LEN(10) DFT(*CURLIB) +
                           SPCVAL((*LIBL) (*CURLIB)) +
                           PROMPT('Bibliothèque')
*-------------------------------------------------------------------------------------------
Utilisation de Liste de paramètre ELEM dans une invite


              PARM       KWD(SELECT) TYPE(LIST1) MIN(1) MAX(35) +
                           PROMPT('Critères de sélection (35)')

  LIST1:      ELEM       TYPE(*CHAR) LEN(15) EXPR(*YES) +
                           PROMPT('Facteur 1')
              ELEM       TYPE(*CHAR) LEN(3) RSTD(*YES) DFT(*EQ) +
                           VALUES('*EQ' '*LE' '*GT' '*LT' '*GE' +
                           '*LE' '*NE' '*CT' '*WC') PROMPT('Opérateur')
              ELEM       TYPE(*CHAR) LEN(15) EXPR(*YES) +
                           PROMPT('Facteur 2')
              ELEM       TYPE(*CHAR) LEN(5) RSTD(*YES) DFT(*NONE) +
                           VALUES('*NONE' 'F1' 'F2') PROMPT('Type +
                           caractère')
              ELEM       TYPE(*NAME) LEN(10) DFT(*NONE) +
                           SPCVAL(('*STD') ('*NONE')) PROMPT('F1 : +
                           Table de translation')
*-------------------------------------------------------------------------------------------
Utilisation de CHOICE pour description des valeurs de choix sur invite de la commande

              PARM      KWD(FIRSTBKP) TYPE(*CHAR) LEN(10) +
                        SPCVAL((NUMERO)) MIN(1) EXPR(*YES) +
               ===>     CHOICE('Numéro instruction') +
                        PROMPT('Premier Breakpoint')

      Utilsation  *GENERIC pour choix de valeur
              PARM       KWD(USER) TYPE(*GENERIC) LEN(10) DFT(*ALL) +
                           SPCVAL((*ALL)) PROMPT('Utilisateur à +
                           Afficher')
*-------------------------------------------------------------------------------------------
Utilisation de fichier MSG pour afficher les prompts de commande (max 30) à créer avec

         CRTCMD  PMTFILE(MSGFIL1/*LIBL)
              CMD        PROMPT(MSG0001)
              PARM       KWD(AUTL) TYPE(*GENERIC) LEN(10) RSTD(*NO) +
                           MIN(1) MAX(1) FILE(*NO) FULL(*NO) +
                           EXPR(*YES) VARY(*NO) PASSATR(*NO) +
                           PROMPT(MSG0002)
```