

## Exercice No. 3

# Créer une application CICS simple

```
TCPIP MSG10 ==> SOURCE DATA SET = SYS1.LOCAL.VTAMLST(USSTCPIP)

02/04/01                W E L C O M E T O                16:54:39

          SSSSSS // 3333333 9999999 0000000
        SS // 33 33 99 99 00 00
       SS // 33 99 99 00 00
      SSSS // 33333 9999999 00 00
     SS // 33 99 00 00
    SS // 33 33 99 99 00 00
   SSSSSS // 3333333 9999999 0000000

YOUR TERMINAL NAME IS :                YOUR IP ADDRESS IS : 217.002.090.033

          APPLICATION DEVELOPMENT SYSTEM
          OS/390 RELEASE 2.7.0

===> ENTER "L " FOLLOWED BY THE APPLID YOU WISH TO LOGON TO.  EXAMPLE "L TSO"
      FOR TSO/E OR "L C001" FOR THE CICSC001 CICS APPLICATION.

l tso
```

Nous voulons écrire un programme CICS simple "Hello World".

Souvenez vous: TSO est système secondaire du OS/390. CICS est un autre système secondaire du OS/390. Chacun des deux systèmes secondaires a sa propre interface (son propre Shell). Pour créer une application CICS, vous devez travailler avec les deux systèmes. Avec TSO pour la construire et avec CICS pour l'exécuter (sous le système secondaire CICS). Puisque l'OS/390 est un système d'exploitation multi-utilisateur (capable de multisessions), Vous pouvez faire fonctionner une session TSO et en même temps une session CICS sous NT sur votre PC. Chaque session fonctionne dans sa propre fenêtre.

Démarez l'émulateur 3270 d'abord pour une session TSO puis plus tard une deuxième fois pour une session CICS.

Commencez avec la session TSO et le processus logon.







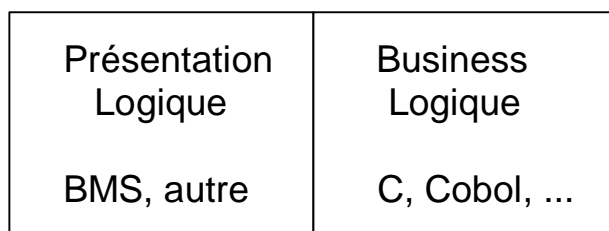
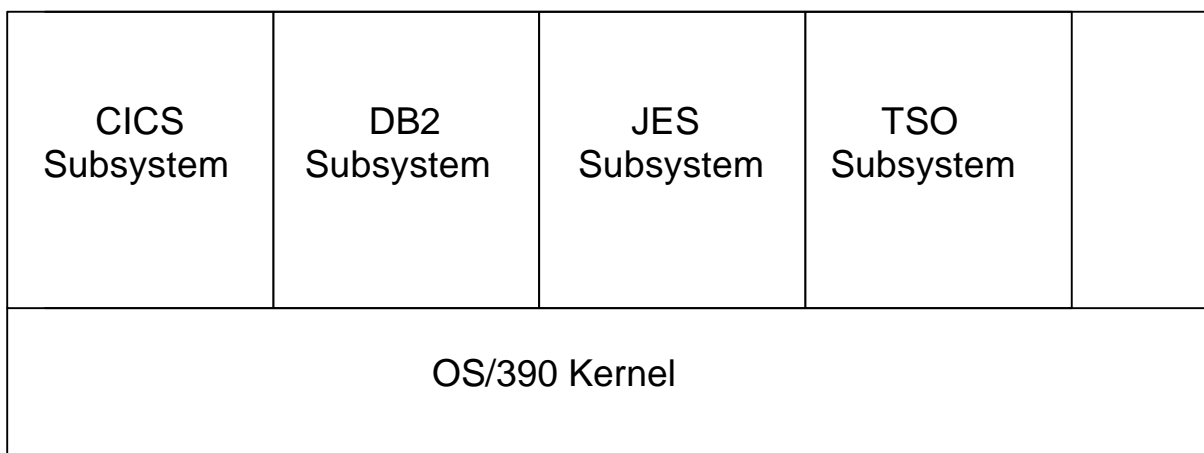
Votre application se compose de deux parties de programme et d'un texte JCL pour la traduction. Nous entrons cette application comme "Members" dans le nouveau "Partitioned Data Set" PRAKT16.CICS.TEST16.

Le développement de programmes CICS permet beaucoup de libertés. Si souhaité, vous pouvez créer au choix des codes sauvages en spagettis.

Un programme CICS structuré correctement se compose de deux parties: business logique et présentation logique. Business logique est la partie dans laquelle sont effectués des calculs et sont lues/écrites des données dans une base de données. Présentation logique est la partie dans laquelle sont représentés les résultats des calculs dans écran agréable pour l'utilisateur.

Business logique est écrits dans un langage comme C++, COBOL, PL/1 etc. Il y a beaucoup d'autres choix pour la présentation logique. Le plus moderne d'entres eux est l'utilisation de pages serveur Java et une application serveur du Web pour représenter le contenu de l'écran dans une "Web browsers". Le plus vieux (et plus simple) choix fait appel au système secondaire CICS BMS (Basic Mapping Support). Les programmes BMS sont écrits dans le langage BMS. Dans notre exemple le Business logique est écrit en C et la présentation logique en BMS.

Commencez avec cette dernière; appelez l'écran "Edit Entry Panel" (voir l'illustration 23 dans notre exemple TSO) et créez un "Member" "MAP16" pour être mis à nouveau dans le "Partitioned Data Set" PRAKT16.CICS.TEST16.



Application CICS

```

File Edit Confirm Menu Utilities Compilers Test Help
#####
VIEW          PRAKT16.CICS.TEST16(MAP16) - 01.05          Columns 00001 00072
***** ***** Top of Data *****
==MSG> -Warning- The UNDO command is not available until you change
==MSG>          your edit profile using the command RECOVERY ON.
000001 //PREPARE JOB ( ),CLASS=A,MSGCLASS=H,MSGLEVEL=(1,1),NOTIFY=&SYSUID
000002 //ASSEM EXEC DFHMAPS,MAPNAME='P16SET',RMODE=24
000003 //SYSUT1 DD *
000004 P16SET DFHMSD TYPE=MAP,MODE=INOUT,LANG=C,STORAGE=AUTO,TIOAPFX=YES
000005 * MENU MAP.
000006 LABEL16 DFHMDI SIZE=(24,80),CTRL=(PRINT,FREEKB)
000007 DFHMDI POS=(9,23),ATTRB=(ASKIP,NORM),LENGTH=34, X
000008 DFHMDI INITIAL='WELCOME TO THE MAGIC WORLD OF CICS'
000009 DFHMDI POS=(12,27),ATTRB=(ASKIP,NORM),LENGTH=26,
000010 INITIAL='MAY THE FORCE BE WITH YOU'
000011 DFHMSD TYPE=FINAL
000012 END
000013 /*
000014 //
***** ***** Bottom of Data *****

Command ==> SUB          Scroll ==> PAGE
F1=Help      F3=Exit      F5=Rfind      F6=Rchange  F12=Cancel

```

Votre programme BMS utilise 3 types d'ordres: DFHMSD, DFHMDI et DFHMDF.

Un écran BMS utilise une résolution 24 x 80 (format d'écran 3270). Les données entrées et leurs résultats sont représentés à l'intérieur d'un champ de matrice 24 x 80 avec les informations suivantes: numéro de ligne, de colonne et longueur du champ. Ceci apparaît à l'aide de la commande DFHMDI. L'ordre DFHMDI à la ligne 000007 définit un champ situé à la ligne 9, qui débute à la colonne 23, d'une longueur de 34 lettres et avec l'inscription "Welcome to the magic World of CICS" initialisée.

Notre exemple de programme BMS contient 2 ordres DFHMDI de ce genre.

L'ordre DFHMSD (ligne 000004) définit un "Mapset" avec le nom "P16SET". Une transaction implique en général plusieurs écrans différents, par exemple un écran dans lequel l'utilisateur peut effectuer une saisie et visualiser sur un autre écran le résultat. Tous les écrans (Maps) d'un type de transaction sont répertoriés dans un "Mapset".

Les différents "Maps" (Screens) d'un "Mapsets" sont définies par l'ordre DFHMDI et sont caractérisées par des étiquettes. Dans votre exemple simple Hello World, le "Mapset" se compose d'une seule Map qui est définie par la désignation "Label16" ligne 000006.

Le "Member" PRAKT16.CICS.TEST16(MAP16) représente en réalité un texte JCL. Contrairement à l'image 29, dans notre exemple TSO les fichiers à traiter ne sont pas indiqués par IN-FILE='xxx.yyy.zzz'. L'ordre JCL à la ligne 000003 "//SYSUT1 DD \*" signifie que le fichier à traiter se situe à la ligne suivante (ligne 000004 à 000012).

Tapez sur la ligne de commande la fonction ISPF "SUB". Cela exporte la procédure DFHMAPS.

JCL trouve le "Member" "DFHMAPS" (ligne 000002) dans la librairie SYS1.PROCLIB(DFHMAPS). Deux fichiers résultent de l'utilisation de l'ordre DFHMAPS. Un contient la traduction du code source BMS qui est placée dans un "Member" dans une "MAPLIB" portant le nom "CICSTS13.CICS.SDFHLOAD". Vous pourrez trouver ici plus tard les composants BMS du système secondaire.



Lors de la création du programme business logique, s'offre à vous la possibilité d'utiliser et de parachever SPRUTH.LIB(P16SET) comme base. Par ceci on garantie (erreurs exclues), que des présentations logiques et des business logiques utilisent des représentations de données identiques.

```

File Edit Confirm Menu Utilities Compilers Test Help
-----
VIEW          SPRUTH.LIB(S04SET) - 01.00                      Columns 00001 00072
***** ***** Top of Data *****
==MSG> -Warning- The UNDO command is not available until you change
==MSG>          your edit profile using the command RECOVERY ON.
000001 union
000002 {
000003 struct {
000004     char          dfhms1Y12";
000005     } label04i;
000006
000007 struct {
000008     char          dfhms2Y12";
000009     } label04o;
000010
000011 } label04;
000012
***** ***** Bottom of Data *****

Command ==>>
F1=Help      F3=Exit      F5=Rfind     F6=Rchange   F12=Cancel

                                Scroll ==>> PAGE

```

Il est possible qu'il apparaisse un "Y" comme dans la représentation du PRAKT16.LIB(P16SET) ci-dessus. Cela doit faire quelque chose avec le "Problème des crochets" lors de la création de programme C et trit z.B. auf, si vous travaillez avec la page de code 037 du OS/390 à la place de la page de code 1047. Les applications de l'OS/390 utilisent fréquemment la page 037. Nous vous recommandons de sélectionner la page code 1047 pour votre client 3270 (émulateur 3270).

Souvenez vous: TSO enregistre toutes ses données dans le format EBCDIC. Dans ce format il y a deux représentations pour les crochets selon la page code utilisée, à savoir

Character	ASCII hex	Vrai EBCDIC hex	Faux EBCDIC hex
Left Square ( [ )	x'5B'	x'AD'	x'BA'
Right Square ( ] )	x'5D'	x'BD'	x'BB'
Character	ASCII hex	Vrai EBCDIC hex	Faux EBCDIC hex
Left Square ( [ )	x'5B'	x'AD'	x'BA'
Right Square ( ] )	x'5D'	x'BD'	x'BB'

L'émulateur 3270 représentera fréquemment hex BA et hex BB par des crochets " [ " et " ] ", tandis que hex AD et hex BD sont représentés par "Y" et " ". Toutefois, il est possible de modifier la représentation avec la plupart des émulateurs 3270 même si dans l'exemple ci-dessus cela n'a pas été fait.

Il est important de comprendre ce problème. Si votre émulateur 3270 n'a pas été reconfiguré, il y a de fortes chances que le problème survienne. La procédure primitive est la suivante:

Lors de la saisie d'un programme les symboles " [ " et " ] " sont employés à leur juste valeur. Cependant, ces derniers sont ensuite faussement enregistrés comme x'BA et x'BB.

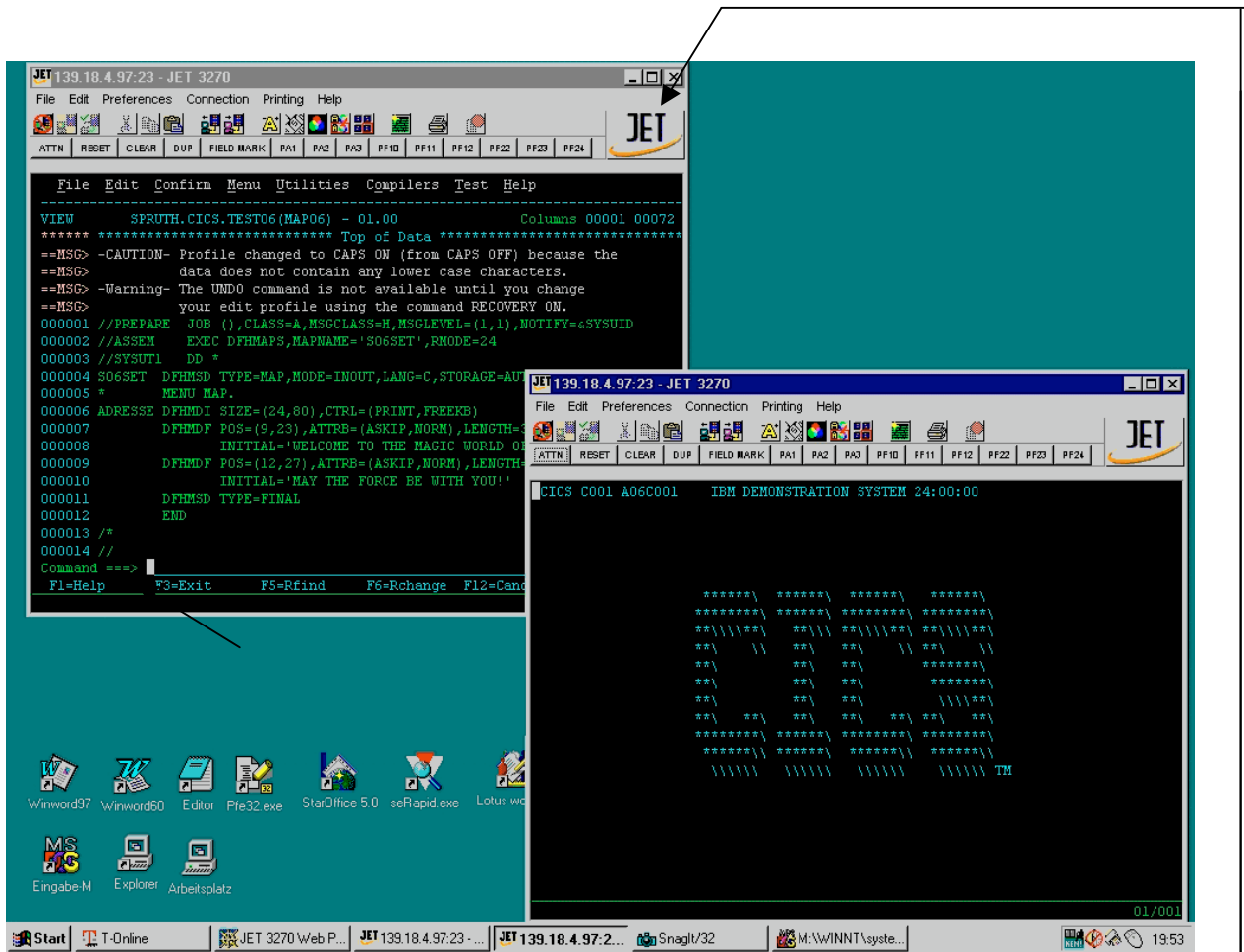


Après avoir entré le programme, deux fonctions ISPF de changement globaux "Change" vous sont proposées en tapant sur la ligne de commande:

```
C [ x'ad' all  
C ] x'bd' all
```

Le symbole "[" est remplacé par la valeur hexadécimale hex "ad". Il en va de même pour le symbole "]" par hex "bd".

"change" peut également être entré au lieu d'un "C".



Développez votre application sous TSO. Mais vous la ferez fonctionner sous CICS. De plus elle devra être en partie installée sous CICS. C'est plus confortable de travailler avec deux sessions OS/390 ouvertes en même temps. La fenêtre située en haut à gauche de l'écran ci-dessus montre une session TSO. Ouvrez une deuxième session OS/390 en rappelant l'émulateur 3270. La deuxième session est représentée ci-dessus par la fenêtre en bas à droite.

Il est possible d'ouvrir une autre session comme précédemment indiqué en utilisant l'émulateur JET3270 en cliquant sur JET dans la partie droite du panneau en haut de la fenêtre. Vous pouvez également obtenir le même résultat avec l'aide de l'Explorateur NT.

Dans le cas d'un "IBM Personal Communications":

>start > Programme > IBM Personal Communications > Début ou configuration de session.

Ils configurent la session de la même manière que la première fois.

Puisque l'OS/390 est un serveur multiutilisateur, ils peuvent utiliser plusieurs sans problème plusieurs sessions en même temps. Alors pourquoi pas ouvrir deux sessions sur un même PC par l'établissement de la même liaison LAN ou ISDN.

TCPIP MSG10 ==> SOURCE DATA SET = SYS1.LOCAL.VTAMLST(USSTCPIP)

02/04/01

W E L C O M E T O

08:55:40

```
SSSSSS // 3333333 9999999 0000000
SS // 33 33 99 99 00 00
SS // 33 99 99 00 00
SSSS // 33333 9999999 00 00
SS // 33 99 00 00
SS // 33 33 99 99 00 00
SSSSSS // 3333333 9999999 0000000
```

YOUR TERMINAL NAME IS :

YOUR IP ADDRESS IS : 217.002.089.066

APPLICATION DEVELOPMENT SYSTEM  
OS/390 RELEASE 2.7.0

==> ENTER "L " FOLLOWED BY THE APPLID YOU WISH TO LOGON TO. EXAMPLE "L TSO"  
FOR TSO/E OR "L C001" FOR THE CICSC001 CICS APPLICATION.

l C001

Procédez au logon et appelez le système secondaire CICS du OS/390 en tapant C001.

CICS C001 A06C001 IBM DEMONSTRATION SYSTEM 24:00:00

```
*****\ *****\ *****\ *****\
*****\ *****\ *****\ *****\
**\\\\**\ **\\\\ **\\\\**\ **\\\\**\
**\ \\ **\ **\ \\ **\ \\
**\ **\ **\ *****\
**\ **\ **\ *****\
**\ **\ **\ \\\\\**\
**\ **\ **\ **\ **\
*****\ *****\ *****\ *****\
*****\ *****\ *****\ *****\
\\\\\\\\ \\\\\\\\\ \\\\\\\\\ \\\\\\\\\ TM
```

L'écran d'accueil CICS apparaît.

Tapez sur la touche Entrer.

```
CICS C001 A06C001      IBM DEMONSTRATION SYSTEM 24:00:00
```

```
*****\  *****\  *****\  *****\  
*****\  *****\  *****\  *****\  
**\\\\\\**\  **\\\\\  **\\\\\\**\  **\\\\\\**\  
**\      \\\  **\    **\      \\\  **\      \\  
**\      \\\  **\    **\      \\\  *****\  
**\      \\\  **\    **\      \\\  *****\  
**\      \\\  **\    **\      \\\  \\\\\\\**\  
**\      **\  **\    **\      **\  **\      **\  
*****\  *****\  *****\  *****\  
*****\  *****\  *****\  *****\  
\\\\\\\\\  \\\\\\\  \\\\\\\  \\\\\\\  TM
```

```
DFHAC2001 02/04/01 11:01:01 A06C001 Transaction '' is not recognized. Check  
that the transaction name is correct.
```

Un message d'erreur s'affiche (sans importance). Déplacez le curseur sur la dernière ligne avec la touche Tab.

```
CICS C001 A06C001      IBM DEMONSTRATION SYSTEM 24:00:00
```

```
*****\  *****\  *****\  *****\  
*****\  *****\  *****\  *****\  
**\\\\\\**\  **\\\\\  **\\\\\\**\  **\\\\\\**\  
**\      \\\  **\    **\      \\\  **\      \\  
**\      \\\  **\    **\      \\\  *****\  
**\      \\\  **\    **\      \\\  *****\  
**\      \\\  **\    **\      \\\  \\\\\\\**\  
**\      **\  **\    **\      **\  **\      **\  
*****\  *****\  *****\  *****\  
*****\  *****\  *****\  *****\  
\\\\\\\\\  \\\\\\\  \\\\\\\  \\\\\\\  TM
```

```
DFHAC2001 02/04/01 11:01:01 A06C001 Transaction '' is not recognized. Check  
that the transaction name is correct. CEDA DISPLAY GROUP(*)
```

CICS attend que vous appeliez une transaction (parmi de nombreuses). Les différentes transactions (applications) sont appelées normalement par la saisie des quatre lettres d'une transaction ID.

L'interpréteur de commande CICS est également utilisé comme transaction. Il est appelé avec la transaction ID "CEDA", suivi par une liste de paramètres qui contient des commandes CICS ainsi que des saisies de données.

Comme exemple tapez la commande "ceda display group(\*)". Puis tapez sur la touche Entrer

```

DISPLAY GROUP(*)
ENTER COMMANDS
GROUP
AOR2TOR
ARTT
ATC
CBPS
CEE
CICREXX
CSQ
CSQCKB
CSQSAMP
CTA1TCP
C001EZA
C001TCP
DBA1
DFH$ACCT
DFH$AFFY
DFH$AFLA
+ DFH$BABR

RESULTS: 1 TO 17
PF 1 HELP      3 END 4 TOP 5 BOT 6 CRSR 7 SBH 8 SFH 9 MSG 10 SB 11 SF 12 CNCL

SYSID=C001 APPLID=A06C001
TIME: 00.00.00 DATE: 01.035

```

Lorsque sont installées des applications (transactions) sous CICS, un "Group" (groupe) est créé pour chaque transaction. Des "Members" comme le programme de l'application lui-même se trouvent dans le "Group", der dazugehörige Mapset sowie ein Eintrag, der die Transaktion mit einer (normalement à 4 positions) TRID (Transactions ID) verknüpft.

La liste des groupes déjà installés est longue de 17 écrans (on peut les faire défiler grâce aux touches F8 et F7).

```

CEDA DEFINE MAPSET(P16SET) GROUP(PR16A)
ENTER COMMANDS
GROUP
AOR2TOR
ARTT
ATC
CBPS
CEE
CICREXX
CSQ
CSQCKB
CSQSAMP
CTA1TCP
C001EZA
C001TCP
DAVIN15
DAVIN4
DAVIN8
DAVIN85
+ DAVIN9

RESULTS: 1 TO 17
PF 1 HELP      3 END 4 TOP 5 BOT 6 CRSR 7 SBH 8 SFH 9 MSG 10 SB 11 SF 12 CNCL

SYSID=C001 APPLID=A06C001
TIME: 10.09.38 DATE: 01.221

```

Définissez pour votre transaction un groupe propre "PR16A" et la dazugehörigen Mapset dans "P16SET". Pour cela, écrivez sur la première ligne, celle qui sert de ligne de commande, l'ordre CEDA suivant: "CEDA DEFINE MAPSET(P16SET) GROUP(PR16A)". Veuillez utiliser des lettres majuscules!

Si vous faites une erreur, vous pouvez appeler l'aide avec la touche F1. Par exemple, vous pouvez supprimer un mauvais groupe entré avec la fonction DELETE. Tapez sur la touche Entrer.

```

CEDA DEFINE MAPSET(P16SET) GROUP(PR16A)
OVERTYPE TO MODIFY                                CICS RELEASE = 0530
CEDA DEFINE Mapset( P16SET  )
  Mapset      : P16SET
  Group       : PR16A
  Description  ==>
  RESident    ==> No                No | Yes
  USAge       ==> Normal            Normal | Transient
  USElpacopy  ==> No                No | Yes
  Status      ==> Enabled            Enabled | Disabled
  RSl         : 00                  0-24 | Public

I New group PR16A created.

DEFINE SUCCESSFUL                                SYSID=C001 APPLID=A06C001
PF 1 HELP 2 COM 3 END                           TIME: 11.27.07 DATE: 01.221
                                                6 CRSR 7 SBH 8 SFH 9 MSG 10 SB 11 SF 12 CNCL

```

CICS vous informe que le nouveau groupe "PR16A" a été créé. Taper à nouveau l'ordre CEDA DISPLAY GROUP(\*), alors vous trouverez dans la liste l'entrée PR16A.

```

CEDA INSTALL GROUP(PR16A)
OVERTYPE TO MODIFY                                CICS RELEASE = 0530
CEDA DEFINE Mapset( P16SET  )
  Mapset      : P16SET
  Group       : PR16A
  Description  ==>
  RESident    ==> No                No | Yes
  USAge       ==> Normal            Normal | Transient
  USElpacopy  ==> No                No | Yes
  Status      ==> Enabled            Enabled | Disabled
  RSl         : 00                  0-24 | Public

I New group PR16A created.

DEFINE SUCCESSFUL                                SYSID=C001 APPLID=A06C001
PF 1 HELP 2 COM 3 END                           TIME: 11.27.07 DATE: 01.221
                                                6 CRSR 7 SBH 8 SFH 9 MSG 10 SB 11 SF 12 CNCL

```

Jusqu'à présent, vous avez communiqué avec le système secondaire CICS et dit qu'un nouveau groupe "PR16A" existait contenant un Mapset "P16SET". Dans la prochaine étape, vous devrez installer PR16A dans la bibliothèque du programme de l'application par le biais de CICS. Ceci est réalisable avec la commande INSTALL de l'interpréteur de lignes de commande CEDA. Elle se situe sur la première ligne.

Tapez sur la touche Entrer.

```

CEDA INSTALL GROUP(PR16A)
OVERTYPE TO MODIFY
CEDA Install
All
Connection ==>
DB2Conn ==>
DB2Entry ==>
DB2Tran ==>
DOctemplate ==>
Enqmodel ==>
File ==>
Journalmodel ==>
LSrpool ==>
Mapset ==>
PARTitionset ==>
PARTner ==>
PROcesstype ==>
PROfile ==>
PROgram ==>
+ Requestmodel ==>

                                     SYSID=C001 APPLID=A06C001
INSTALL SUCCESSFUL                       TIME: 11.33.41 DATE: 01.221
PF 1 HELP                               3 END          6 CRSR 7 SBH 8 SFH 9 MSG 10 SB 11 SF 12 CNCL

```

CEDA vous informe que l'installation a été réussie.

Changer de fenêtre TSO et appelez le "Member" PRAKT16.LIB(P16SET).

```

File Edit Confirm Menu Utilities Compilers Test Help
ssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssss
VIEW      PRAKT16.LIB(P16SET) - 01.00                                     Columns 00001 00072
***** Top of Data *****
==MSG> -CAUTION- Data contains invalid (non-display) characters. Use command
==MSG>          ==> FIND P'.' to position cursor to these
==MSG> -Warning- The UNDO command is not available until you change
==MSG>          your edit profile using the command RECOVERY ON.
000001 union
000002 {
000003 struct {
000004     char          dfhms1 12 ;
000005     } label16i;
000006
000007 struct {
000008     char          dfhms2 12 ;
000009     } label16o;
000010
000011 } label16;
000012
***** Bottom of Data *****
Command ==>          Scroll ==> PAGE
F1=Help     F3=Exit      F5=Rfind      F6=Rchange    F12=Cancel

```

Bei der Übersetzung des MAP BMS Quellcodes wurde für uns als Nebenprodukt im Member PRAKT16.LIB(P16SET) ein Template für unser Anwendungsprogramm erzeugt. Wir kopieren den Member in einen neuen Member PROG16 uderes Partitioned Data Sets PRAKT16.CICS.TEST16.

Tapez sur la touche F3.

```

Menu  Functions  Confirm  Utilities  Help
#####
VIEW          PRAKT16.LIB          Row 00001 of 00001
Name          Prompt          VV MM    Changed    Size  Init  Mod  ID
---c-----  P16SET          *Viewed
**End**

Command ==>
F1=Help      F3=Exit      F10=Actions  F12=Cancel

Scroll ==> PAGE

```

Copiez ce "Member" en tapant c (copy) devant le nom du "Member".

Tapez sur la touche Entrer.

```

RefList  Help
#####
COPY Entry Panel

CURRENT from data set: 'PRAKT16.LIB(P16SET)'

To Library          Replace option:
Project . . . PRAKT16      Enter "/" to select option
Group . . . CICS          Replace like-named members
Type . . . TEST16

To Other Data Set Name
Data Set Name . . .
Volume Serial . . . (If not cataloged)

NEW member name . . . PROG16 (Blank unless member to be renamed)

Options
Sequential Disposition  Pack Option          SCLM Setting
2 1. Mod                1 1. Default        3 1. SCLM
2. Old                  2. Pack              2. Non-SCLM
                        3. As is

Command ==>
F1=Help      F3=Exit      F12=Cancel

```

Wir geben als Ziel für den Kopiervorgang das Member PROG16 innerhalb unseres Partitioned Data Sets Prakt16.CICS.TEST16 an. Dieses Member soll unser C Programm enthalten.

Tapez sur la touche Entrer.



```

Menu  Functions  Confirm  Utilities  Help
#####
VIEW          PRAKT16.LIB                      Row 0001 of 0001

  Name      Prompt          VV MM   Changed  Size  Init  Mod  ID
-----    -
P16SET    *Copied
**End**

Command ==>
F1=Help   F3=Exit   F10=Actions  F12=Cancel                     Scroll ==> PAGE
  
```

Le message "\*copied" apparaît. Wir schauen uns das Member SPRUTH.CICS.TEST04(PROG04) an.

Tapez sur la touche F3.

```

Menu  Options  View  Utilities  Compilers  Help
#####
DSLIST - Data Sets Matching PRAKT16                      Row 1 of 8

Command - Enter "/" to select action          Message          Volume
-----
PRAKT16                                     *ALIAS
PRAKT16.CICS.TEST16                        SMS001
PRAKT16.ISPF.ISPPROF                      SMS001
PRAKT16.LIB                               SMS001
PRAKT16.SPFLOG1.LIST                      SMS001
PRAKT16.TEST.C                           SMS001
PRAKT16.TEST.CNTL                        SMS001
PRAKT16.TEST.LOAD                         SMS001
***** End of Data Set list *****

v
Command ==>
F1=Help   F3=Exit   F5=Rfind   F12=Cancel                     Scroll ==> PAGE
  
```

V (View) devant la saisie PRAKT16.CICS.TEST16

Tapez sur la touche Entrer.

A l'écran suivant, tapez à nouveau un "v" devant la saisie PROG16.

Tapez à nouveau sur la touche Entrer.

```

File Edit Confirm Menu Utilities Compilers Test Help
ssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssss
VIEW          PRAKT16.CICS.TEST16(PROG16) - 01.00                Columns 00001 00072
***** ***** Top of Data *****
==MSG> -CAUTION- Data contains invalid (non-display) characters. Use command
==MSG>          ==> FIND P '.' to position cursor to these
==MSG> -Warning- The UNDO command is not available until you change
==MSG>          your edit profile using the command RECOVERY ON.
000001 union
000002 {
000003 struct {
000004     char          dfhms1 12 ;
000005     } label16i;
000006
000007 struct {
000008     char          dfhms2 12 ;
000009     } label16o;
000010
000011 } label16;
000012
***** ***** Bottom of Data *****

Command ==>
F1=Help      F3=Exit      F5=Rfind      F6=Rchange  F12=Cancel
Scroll ==> PAGE

```

Le template occupe les lignes 000001 à 000012.

Taper 5 fois sur la touche F3 vous ramène à l'écran "ISPF Primary Option". Tapez "2" puis sur la touche Entrer.

```

Menu RefList RefMode Utilities LMF Workstation Help
ssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssss
                        Edit Entry Panel

ISPF Library:
  Project . . . PRAKT16
  Group . . . . CICS      . . .      . . .      . . .
  Type . . . . TEST16
  Member . . . PROG16      (Blank or pattern for member selection list)


Other Partitioned or Sequential Data Set:
  Data Set Name . . .
  Volume Serial . . .      (If not cataloged)

Workstation File:
  File Name . . .

Options
Initial Macro . . . . / Confirm Cancel/Move/Replace
Profile Name . . . . Mixed Mode
Format Name . . . . Edit on Workstation
Data Set Password . . . Preserve VB record length

Command ==>
F1=Help      F3=Exit      F10=Actions  F12=Cancel

```



Tapez PRAKT16.CICS.TEST16(PROG16) comme Data Set à modifier.

Tapez sur la touche Entrer.

```

File Edit Confirm Menu Utilities Compilers Test Help
ssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssss
EDIT      PRAKT16.CICS.TEST16(PROG16) - 01.00              Columns 00001 00072
*****  ***** Top of Data *****
==MSG> -CAUTION- Data contains invalid (non-display) characters. Use command
==MSG>      ==> FIND P'.' to position cursor to these
==MSG> -Warning- The UNDO command is not available until you change
==MSG>      your edit profile using the command RECOVERY ON.
==MSG> -CAUTION- Profile is set to STATS ON. Statistics did not exist for
==MSG>      this member, but will be generated if data is saved.
000001 union
000002 {
000003 struct {
000004     char          dfhms1 12 ;
000005     } label16i;
000006
000007 struct {
000008     char          dfhms2 12 ;
000009     } label16o;
000010
000011 } label16;
00i012
Command ==>
F1=Help      F3=Exit      F5=Rfind      F6=Rchange  F12=Cancel      Scroll ==> PAGE

```

Wir erstellen unser CICS Anwendungsprogramm in C, indem wir das hier vorliegende Gerüst vervollständigen. An die vorliegende Union Deklaration hängen wir die eigentliche main Routine an. (Wir erinnern uns, eine Union in C ist eine Variable, die Objekte unterschiedliche Größe und Typs enthalten kann - hier zwei Strukturen).

Tapez "i" (Insert) sur la ligne située dessous die union.

Tapez sur la touche Entrer.

```

File Edit Confirm Menu Utilities Compilers Test Help
ssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssss
EDIT      PRAKT16.CICS.TEST16(PROG16) - 01.00              Columns 00001 00072
==MSG> -CAUTION- Data contains invalid (non-display) characters. Use command
==MSG>      ==> FIND P'.' to position cursor to these
==MSG> -Warning- The UNDO command is not available until you change
==MSG>      your edit profile using the command RECOVERY ON.
==MSG> -CAUTION- Profile is set to STATS ON. Statistics did not exist for
==MSG>      this member, but will be generated if data is saved.
000001 union
000002 {
000003 struct {
000004     char          dfhms1 12 ;
000005     } label16i;
000006
000007 struct {
000008     char          dfhms2 12 ;
000009     } label16o;
000010
000011 } label16;
000012
'''' main()
Command ==>
F1=Help      F3=Exit      F5=Rfind      F6=Rchange  F12=Cancel      Scroll ==> PAGE

```

Tapez à présent le premier ordre de votre programme C sur la ligne vièrge.

Tapez sur la touche Entrer.

```

File Edit Confirm Menu Utilities Compilers Test Help
ssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssss
EDIT          PRAKT16.CICS.TEST16(PROG16) - 01.00          Columns 00001 00072
==MSG> -CAUTION- Profile is set to STATS ON. Statistics did not exist for
==MSG>          this member, but will be generated if data is saved.
000001 union
000002 {
000003 struct {
000004     char          dfhms1 12 ;
000005     } label16i;
000006
000007 struct {
000008     char          dfhms2 12 ;
000009     } label16o;
000010
000011 } label16;
000012
000013 main()
000014 {
000015     EXEC CICS SEND MAP("label16") MAPSET("p16set") ERASE;
000016     '.....'
***** ***** Bottom of Data *****
Command ===>
          F1=Help          F3=Exit          F5=Rfind          F6=Rchange          F12=Cancel          Scroll ===> PAGE

```

Wir vervollständigen unser Programm. Es besteht aus einem einzigen Statement

```
EXEC CICS SEND MAP("label16") MAPSET("p16set") ERASE;
```

Dieser Befehl bewirkt, daß die BMS Map "label16" aus dem Mapset "p16set" an den 3270 Terminal gesendet wird.

En actionnant la touche F3 vous enregistrez votre programme.

```

Menu RefList RefMode Utilities LMF Workstation Help
ssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssss
          Edit Entry Panel          Member PROG16 saved

ISPF Library:
Project . . . . PRAKT16
Group . . . . CICS . . . . .
Type . . . . TEST16
Member . . . . PROG16          (Blank or pattern for member selection list)

Other Partitioned or Sequential Data Set:
Data Set Name . . .
Volume Serial . . .          (If not cataloged)

Workstation File:
File Name . . . .

Initial Macro . . . .          Options
Profile Name . . . .          / Confirm Cancel/Move/Replace
Format Name . . . .          Mixed Mode
Data Set Password . . . .     Edit on Workstation
                               Preserve VB record length

Command ===>
          F1=Help          F3=Exit          F10=Actions          F12=Cancel

```

Wir geben PROG16 ein, um uns das Ergebnis nochmals anzusehen.

Tapez sur la touche Entrer.

```

File Edit Confirm Menu Utilities Compilers Test Help
ssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssss
EDIT          PRAKT16.CICS.TEST16(PROG16) - 01.00          Columns 00001 00072
***** ***** Top of Data *****
==MSG> -CAUTION- Data contains invalid (non-display) characters. Use command
==MSG>          ===> FIND P'.' to position cursor to these
==MSG> -Warning- The UNDO command is not available until you change
==MSG>          your edit profile using the command RECOVERY ON.
000001 union
000002 {
000003 struct {
000004     char          dfhms1 12 ;
000005     } label16i;
000006
000007 struct {
000008     char          dfhms2 12 ;
000009     } label16o;
000010
000011 } label16;
000012
000013 main()
000014 {
Command ===>
          F1=Help          F3=Exit          F5=Rfind          F6=Rchange          F12=Cancel          Scroll ===> PAGE

```

Vôtre programme erstreckt sich über mehr als einen Bildschirm. Mit der F8 Taste können wir uns Programmzeilen 000015 und 000016 auf dem nächsten Bildschirm ansehen, ...

```

File Edit Confirm Menu Utilities Compilers Test Help
ssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssss
EDIT          PRAKT16.CICS.TEST16(PROG16) - 01.00          Columns 00001 00072
000015     EXEC CICS SEND MAP("label16") MAPSET("p16set") ERASE;
000016 }
***** ***** Bottom of Data *****

Command ===>
          F1=Help          F3=Exit          F5=Rfind          F6=Rchange          F12=Cancel          Scroll ===> PAGE

```

Et en actionnant la touche F7 vous revenez en arrière.

En actionnant la touche F3 vous retournez à l'écran "Edit Entry".



```
File Edit Confirm Menu Utilities Compilers Test Help
ssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssss
EDIT PRAKT16.CICS.TEST16(START16) - 01.03 Columns 00001 00072
***** ***** Top of Data *****
==MSG> -CAUTION- Profile changed to NUMBER ON STD (from NUMBER OFF).
==MSG> Data has valid standard numbers.
==MSG> -CAUTION- Profile changed to CAPS ON (from CAPS OFF) because the
==MSG> data does not contain any lower case characters.
==MSG> -Warning- The UNDO command is not available until you change
==MSG> your edit profile using the command RECOVERY ON.
000010 //CICSPRE JOB (),CLASS=A,MSGCLASS=H,MSGLEVEL=(1,1),NOTIFY=&SYSUID,
000020 // TIME=1440
001100 // EXEC PROC=CTOCICS,REG=0M ←
001300 //TRN.SYSIN DD DISP=SHR,DSN=PRAKT16.CICS.TEST16(PROG16)
001400 //LKED.SYSIN DD *
001500 NAME PROG16(R)
***** ***** Bottom of Data *****

Command ==> sub Scroll ==> PAGE
F1=Help F3=Exit F5=Rfind F6=Rchange F12=Cancel
```

Zur Übersetzung wird für den Partitioned Data Set "PRAKT16.CICS.TEST16" ein neues Member "START16" erstellt. Dies ist eine JCL File, welche die Prozedur CTOCICS (Compile to CICS) enthält. CTOCICS ruft zunächst den CICS Precompiler auf, der alle CICS Befehle in C Befehle übersetzt. Anschließend wird der C Compiler aufgerufen, der ein Maschinenprogramm erstellt und in eine für das CICS Subsystem zugängliche Library stellt. START16 wird mit dem Kommando "sub" ausgeführt.

Tapez Sub puis sur la touche Entrer.

```
IKJ56250I JOB CICSPRE(JOB04769) SUBMITTED
***
```

Wir warten, bis der Job ausgeführt wurde.

Tapez sur la touche Entrer.

```
18.01.00 JOB04769 $HASP165 CICS PRE ENDED AT N1 MAXCC=4 CN(INTERNAL)
***
```

Ausführung erfolgreich.

Als nächsten Schritt wechseln wir wieder von der TSO Session Fenster zum CICS Session Fenster. Wir nehmen an, es ist noch vom letzten Mal her offen. Wenn nicht, öffnen Sie erneut eine CICS CEDA Session: lc001, ceda display group(\*) .

```
CEDA DEFINE PROGRAM(PROG16) GROUP(PR16A)
ENTER COMMANDS
GROUP
AOR2TOR
ARTT
ATC
CBPS
CEE
CICREXX
CSQ
CSQCKB
CSQSAMP
CTA1TCP
C001EZA
C001TCP
DAVIN15
DAVIN4
DAVIN8
DAVIN85
+ DAVIN9

RESULTS: 1 TO 17
PF 1 HELP          3 END 4 TOP 5 BOT 6 CRSR 7 SBH 8 SFH 9 MSG 10 SB 11 SF 12 CNCL

SYSID=C001 APPLID=A06C001
TIME: 18.18.24 DATE: 01.221
```

Wir definieren für die Gruppe PR16A unser Anwendungsprogramm als PROG16, indem wir den entsprechenden CEDA Befehl in die oberste Zeile schreiben.

Tapez sur la touche Entrer.



```

CEDA DEFINE PROGRAM(PROG16) GROUP(PR16A)
OVERTYPE TO MODIFY                                CICS RELEASE = 0530
CEDA DEFine PROGram( PROG16 )
  PROGram      : PROG16
  Group        : PR16A
  DDescription ==>
  Language     ==> CObol | Assembler | Le370 | C | Pli
  REload       ==> No      No | Yes
  RESident     ==> No      No | Yes
  USAge        ==> Normal  Normal | Transient
  USElpacopy   ==> No      No | Yes
  Status       ==> Enabled  Enabled | Disabled
  RSl          : 00        0-24 | Public
  CEdf         ==> Yes     Yes | No
  DAtallocation ==> Below  Below | Any
  EXEckey     ==> User    User | Cics
  COncurrency  ==> Quasirent Quasirent | Threadsafe
REMOTE ATTRIBUTES
  DYNAMIC      ==> No      No | Yes
+ REMOTESystem ==>

                                SYSID=C001 APPLID=A06C001
DEFINE SUCCESSFUL                TIME: 18.30.52 DATE: 01.221
PF 1 HELP 2 COM 3 END            6 CRSR 7 SBH 8 SFH 9 MSG 10 SB 11 SF 12 CNCL

```

CEDA will einiges von uns wissen. Wir übernehmen alle Default Werte und geben zusätzlich als Sprache Le370 an.

```

CEDA DEFINE PROGRAM(PROG16) GROUP(PR16A)
OVERTYPE TO MODIFY                                CICS RELEASE = 0530
CEDA DEFine PROGram( PROG16 )
  PROGram      : PROG16
  Group        : PR16A
  DDescription ==>
  Language     ==> LE370    CObol | Assembler | Le370 | C | Pli
  REload       ==> No      No | Yes
  RESident     ==> No      No | Yes
  USAge        ==> Normal  Normal | Transient
  USElpacopy   ==> No      No | Yes
  Status       ==> Enabled  Enabled | Disabled
  RSl          : 00        0-24 | Public
  CEdf         ==> Yes     Yes | No
  DAtallocation ==> Below  Below | Any
  EXEckey     ==> User    User | Cics
  COncurrency  ==> Quasirent Quasirent | Threadsafe
REMOTE ATTRIBUTES
  DYNAMIC      ==> No      No | Yes
+ REMOTESystem ==>

                                SYSID=C001 APPLID=A06C001
DEFINE SUCCESSFUL                TIME: 18.30.52 DATE: 01.221
PF 1 HELP 2 COM 3 END            6 CRSR 7 SBH 8 SFH 9 MSG 10 SB 11 SF 12 CNCL

```

Was ist denn Le370 für eine Sprache ? Nun, Le370 ist überhaupt keine Sprache, sondern eine Laufzeitumgebung. CICS braucht an dieser Stelle in Wirklichkeit nicht die Angabe der Quellsprache unseres Anwendungsprogramms (wir haben es ja bereits übersetzt), sondern die Angabe der Laufzeitumgebung des von uns verwendeten Compilers. Alle modernen OS/390 Compiler verwenden eine gemeinsame Laufzeitumgebung, die auf den Namen Le370 getauft wurde.

Tapez sur la touche Entrer.

```

OVERTYPE TO MODIFY                                CICS RELEASE = 0530
CEDA DEFine PROGram( PROG16  )
  PROGram      : PROG16
  Group        : PR16A
  DDescription ==>
  Language     ==> Le370           CObol | Assembler | Le370 | C | Pli
  REload       ==> No             No | Yes
  RESident     ==> No             No | Yes
  USAge        ==> Normal         Normal | Transient
  USElpacopy   ==> No             No | Yes
  Status       ==> Enabled        Enabled | Disabled
  RSl          : 00              0-24 | Public
  CEdf         ==> Yes            Yes | No
  DAllocation ==> Below          Below | Any
  EXEckey     ==> User           User | Cics
  COncurrency ==> Quasirent       Quasirent | Threadsafe
  REMOTE ATTRIBUTES
  Dynamic      ==> No            No | Yes
+ REMOTESystem ==>

                                           SYSID=C001 APPLID=A06C001
  DEFINE SUCCESSFUL                       TIME: 18.41.21 DATE: 01.221
PF 1 HELP 2 COM 3 END                     6 CRSR 7 SBH 8 SFH 9 MSG 10 SB 11 SF 12 CNCL

```

Es erscheint der obige Bildschirm

Tapez sur la touche F3

```

CEDA CEDA DEFINE PROGRAM(PROG16) GROUP(PR16A)
STATUS:  SESSION ENDED

```

Tapez sur la première ligne l'ordre CEDA

```

CEDA DEFINE TRANS(P16A) GROUP(PR16A)
STATUS:▲ SESSION ENDED

```

Unsere Transaktion soll wie alle anderen Transaktion vom Bildschirm über eine 4-stellige Transaktions ID aufgerufen werden. Wir wählen hierfür die ID „P16A“ und teilen diese Wahl mit Hilfe des CEDA DEFINE Befehls mit. Genauso wie PROG16 wird dies Bestandteil von GROUP(PR16A).

Tapez sur la touche Entrer.

```

DEFINE TRANS(P16A) GROUP(PR16A)
OVERTYPE TO MODIFY                                CICS RELEASE = 0530
  CEDA DEFine TRANSAction( P16A )
  TRANSAction ==> P16A
  Group       ==> PR16A
  Description ==>
  PROGram    ==>
  TWAsize    ==> 00000                0-32767
  PROFile    ==> DFHCICST
  PArtitionset ==>
  STatus     ==> Enabled              Enabled | Disabled
  PRIMedsize : 00000                0-65520
  TASKDATAloc ==> Below              Below | Any
  TASKDATAKey ==> User                User | Cics
  STOrageclear ==> No                 No | Yes
  RUNaway     ==> System              System | 0 | 500-2700000
  SHutdown    ==> Disabled            Disabled | Enabled
  ISolate     ==> Yes                 Yes | No
  Brexit      ==>
+ REMOTE ATTRIBUTES
  S PROGRAM OR REMOTESYSTEM MUST BE SPECIFIED.
                                           SYSID=C001 APPLID=A06C001
PF 1 HELP 2 COM 3 END                    6 CRSR 7 SBH 8 SFH 9 MSG 10 SB 11 SF 12 CNCL

```

CEDA will mehrere Angaben von uns und schlägt eine Reihe von Default Werten vor.

```

DEFINE TRANS(P16A) GROUP(PR16A)
OVERTYPE TO MODIFY                                CICS RELEASE = 0530
CEDA DEFINE TRANSACTION( P16A )
TRANSACTION ==> P16A
GROUP ==> PR16A
DESCRIPTION ==>
PROGRAM ==> PROG16
TWSIZE ==> 00000                                0-32767
PROFILE ==> DFHCICST
PARTITIONSET ==>
STATUS ==> Enabled                             Enabled | Disabled
PRIMESIZE : 00000                               0-65520
TASKDATALOC ==> Below                          Below | Any
TASKDATAKEY ==> User                           User | Cics
STORAGECLEAR ==> No                            No | Yes
RUNAWAY ==> System                             System | 0 | 500-2700000
SHUTDOWN ==> Disabled                          Disabled | Enabled
ISOLATE ==> Yes                                Yes | No
BREXIT ==>
+ REMOTE ATTRIBUTES
S PROGRAM OR REMOTESYSTEM MUST BE SPECIFIED.
SYSID=C001 APPLID=A06C001

PF 1 HELP 2 COM 3 END                            6 CRSR 7 SBH 8 SFH 9 MSG 10 SB 11 SF 12 CNCL

```

Wir übernehmen sie alle und geben in die Zeile "PROGRAM" den Namen unseres Anwendungsprogramms, nämlich "PROG16" ein.

Tapez sur la touche Entrer.

```

OVERTYPE TO MODIFY                                CICS RELEASE = 0530
CEDA DEFINE TRANSACTION( P16A )
TRANSACTION : P16A
GROUP : PR16A
DESCRIPTION ==>
PROGRAM ==> PROG16
TWSIZE ==> 00000                                0-32767
PROFILE ==> DFHCICST
PARTITIONSET ==>
STATUS ==> Enabled                             Enabled | Disabled
PRIMESIZE : 00000                               0-65520
TASKDATALOC ==> Below                          Below | Any
TASKDATAKEY ==> User                           User | Cics
STORAGECLEAR ==> No                            No | Yes
RUNAWAY ==> System                             System | 0 | 500-2700000
SHUTDOWN ==> Disabled                          Disabled | Enabled
ISOLATE ==> Yes                                Yes | No
BREXIT ==>
+ REMOTE ATTRIBUTES
SYSID=C001 APPLID=A06C001
TIME: 18.56.09 DATE: 01.221
DEFINE SUCCESSFUL
PF 1 HELP 2 COM 3 END                            6 CRSR 7 SBH 8 SFH 9 MSG 10 SB 11 SF 12 CNCL

```

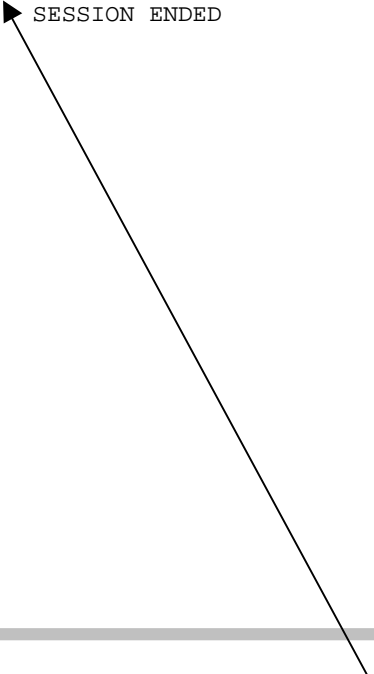
Le message "DEFINE SUCCESSFUL" apparaît.

Tapez sur la touche F3.

```
CEDA DEFINE TRANS(P16A) GROUP(PR16A)
STATUS:  SESSION ENDED
```

Der obige Bildschirm erscheint. Wir haben den Namen unseres Anwendungsprogramms und eine dazugehörige Transaktions ID an CICS bekanntgegeben. Jetzt müssen diese in die CICS Programm-bibliothek übernommen (installiert) werden.

```
CEDA INSTALL GROUP(PR16A)
STATUS:  SESSION ENDED
```



Tapez sur la commande CEDA INSTALL sur la première ligne.

Tapez sur la touche Entrer.

```

INSTALL GROUP(PR16A)
OVERTYPE TO MODIFY
CEDA Install
All
Connection ==>
DB2Conn ==>
DB2Entry ==>
DB2Tran ==>
DOctemplate ==>
Enqmodel ==>
File ==>
Journalmodel ==>
LSrpool ==>
Mapset ==>
PARTitionset ==>
PARTner ==>
PROcesstype ==>
PROfile ==>
PROgram ==>
+ Requestmodel ==>

                                SYSID=C001 APPLID=A06C001
INSTALL SUCCESSFUL             TIME: 19.01.55 DATE: 01.221
PF 1 HELP                      3 END                6 CRSR 7 SBH 8 SFH 9 MSG 10 SB 11 SF 12 CNCL

```

CEDA bestätigt, daß die Installation erfolgreich war.

Tapez sur la touche F3

```

CEDA INSTALL GROUP(PR16A)
STATUS:  SESSION ENDED

```

Das war es. Der obige Bildschirm erscheint. Unsere Transaktion ist als Teil der CICS Anwendungsbi-  
bliothek installiert worden und kann nun aufgerufen und damit ausgeführt werden.

Hierzu löschen wir die oberste Zeile (die CEDA Kommando Zeile) ganz, und rufen unsere Anwendung  
auf, indem wir eben dort unsere Transaktions ID, nämlich P16A, eingeben.

```
WELCOME TO THE MAGIC WORLD OF CICS
```

```
MAY THE FORCE BE WITH YOU!
```

Und hier ist der Erfolg unserer Bemühungen. Eine echte CICS Transaktion, auch wenn sie nicht sehr viel leistet.

Tapez sur la touche Entrer.

```
WELCOME TO THE MAGIC WORLD OF CICS
```

```
MAY THE FORCE BE WITH YOU!
```

```
DFHAC2001 02/04/01 11:31:55 A06C001 Transaction '' is not recognized. Check  
that the transaction name is correct.
```

Dies terminiert die Bildschirmausgabe unserer Transaktion und erzeugt wieder eine (belanglose) Fehlermeldung.

\_WELCOME TO THE MAGIC WORLD OF CICS

MAY THE FORCE BE WITH YOU!

DFHAC2001 02/04/01 11:31:55 A06C001 Transaction '' is not recognized. Check that the transaction name is correct. CEDA DISPLAY GROUP(PR16A)

Alle Bestandteile unserer Transaktion sind in der Gruppe SPRUTH4 gespeichert. Wir schauen sie uns an.

Tapez sur la touche Entrer.

DISPLAY GROUP(PR16A)

ENTER COMMANDS

NAME	TYPE	GROUP	DATE	TIME
P16SET	MAPSET	PR16A	01.221	11.27.06
PROG16	PROGRAM	PR16A	01.221	18.41.21
P16A	TRANSACTION	PR16A	01.221	18.56.08

RESULTS: 1 TO 3 OF 3  
PF 1 HELP 3 END 4 TOP 5 BOT 6 CRSR 7 SBH 8 SFH 9 MSG 10 SB 11 SF 12 CNCL

SYSID=C001 APPLID=A06C001  
TIME: 19.15.24 DATE: 01.221

Die Gruppe PR16A besteht aus den drei Komponenten P16SET, PROG16 und P16A, die wir unter CICS definiert und anschließend installiert haben.

Das war es. Mit HTML wäre es sicher schneller gegangen. Aber wir haben die Basis geschaffen, wirklich leistungsfähige CICS Transaktionen zu erzeugen, etwas, was mit HTML nicht möglich ist.

The End