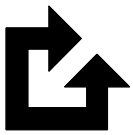
SQT1-2185-00



QPAC for Batch and CICS Online

Installation and System Administration

Version 8 Release 5

First Edition (December 2009)

This edition applies to version 8 release 5 of the Osys Software AG licensed program QPAC for Batch and CICS online, program number 8050-QP-85-T10 and to all subsequent versions, releases and modifications until otherwise indicated in new editions.
Consult Osys Software AG for current information on this product.

Order publications or address your comments to one of the following addresses

Osys Software AG
Hofwiesenstrasse 350, CH-8050 Zurich/Switzerland
Telephone +41 44 317 18 19
Telefax +41 44 317 18 20
E-Mail office@osys.ch

Osys AG
Hofwiesenstrasse 350, CH-8050 Zurich/Switzerland
Telephone +41 44 317 18 19
Telefax +41 44 317 18 20
E-Mail office@osys.ch

© Copyright Osys Software AG 2009. All rights reserved.

Special Notices

The terms listed below are used in this publication and are trademarks or service marks of the following companies:

Osys Software AG Zurich, Switzerland:
Osys-QPAC

International Business Machines:
z/OS
CICS
VSAM
DL/I
DB2
RACF
MQSeries
ISPF/PDF-TSO

Contents

Figures

Chapter 1. Product Installation

Installation Process for Electronic E-Mail Distribution

QPAC will now be distributed in electronic format by e-mail.

To install QPAC you have to do a binary upload of the attachment to your host. You then have to unpack it with the IBM utility TRSMAIN.
If needed you can download the utility from the IBM web site.

In the following section a possible sequence is described how to install QPAC on your system.

1. Copy the attachment to your PC, e.g. using the right mouse button and then "save as".

2. Transfer the saved file as binary to the host.
   Pay attention that the file will receive the attributes RECFM=FB and LRECL=1024 on the host, otherwise the utility TRSMAIN will not be able to read it.

   With FTP take the following steps:
   - Open a command prompt window.
   - Change to the directory where the file has been stored.
   - Enter the command "ftp <YourHostName>".
   - Answer the questions for userid and password.
   - You should now get an ftp> prompt.
   - Enter the command "**binary**".
   - Enter the command "**quote site recfm=fb lrecl=1024**".
   - Enter the command "**put <YourPCFileName> <YourHostFileName>**"
     <YourPCFileName> must correspond to the name of the attachment.
     <YourHostFileName> is the name of the file on the host.
     Attention: If you specify the host file name without quotes, your userid will be taken as HLQ.
   - Enter the qommand "**quit**" to terminate the FTP connection.

3. Now unpack the transferred file with the TRSMAIN program.

   Following a sample step:

```
 //STEP     EXEC PGM=TRSMAIN,PARM=UNPACK
 //STEPLIB  DD DISP=SHR,DSN=<where.TRSMAIN.lives>
//SYSPRINT DD SYSOUT=*
//INFILE   DD DISP=SHR,DSN=<yourHostFilename>
//OUTFILE  DD DISP=(,CATLG,DELETE),UNIT=SYSDA,
//            SPACE=(CYL,(5,1,30)),
//            DSN=<Name.of.unpacked.file>
```

Fig. 1: TRSMAIN

   The resulting PDS contains the following members:
   - QPACINFO Sample program for Type 3 SVC 235
   - QPACPROG QPAC objects for the linkage step
   - DBV7BDB2 nur mit DB2 Support; DB2 V7 Version des QPACBDB2
   - DBV7DBRM nur mit DB2 Support; DB2 V7 DBRM fuer den Bind
   - DBV8BDB2 only with DB2 support; DB2 V8 version of QPACBDB2
   - DBV8DBRM only with DB2 support; DB2 V8 DBRM for the bind

4. The installation of QPAC can now begin.
   Therefore process the member QPACPROG by the linkage editor:

```
//LKED     EXEC PGM=HEWLH096,
               PARM=(XREF,LET,LIST,MAP),REGION=6144K
//SYSUT1   DD  UNIT=SYSALLDA,SPACE=(1024,(200,20))
//SYSLMOD  DD  DISP=SHR,DSN=<QPAC.linklib>
//SYSLIN   DD  DISP=SHR,
               DSN=<Name.of.unpacked.file>(QPACPROG)
//SYSLIB   DD  DSN=DSNxxx.SDSNLOAD,DISP=SHR            DB2 Support
//         DD  DSN=IMSxxx.SDFSRESL,DISP=SHR            IMS Support
//         DD  DSN=CSQxxx.SCSQLOAD,DISP=SHR            MQS Support
//         DD  DSN=CICSTSxx.CICS.SDFHEXCI,DISP=SHR     CICS Support
//SYSPRINT DD  SYSOUT=*
```

Fig. 2: Linkage Editor

For DB2 support the following additional actions must be taken:
DB2 version 7:

> For DB2 version 7 support the corresponding DB2 version 7 QPAC modules must be installed.
  Therefore process member DBV7BDB2 by the linkage editor.

```
//LKED     EXEC PGM=HEWLH096,PARM=(XREF,LET,LIST,MAP),
//              REGION=6144K
//SYSUT1   DD  UNIT=SYSALLDA,SPACE=(1024,(200,20))
//SYSLMOD  DD  DISP=SHR,DSN=<QPAC.linklib>
//SYSLIN   DD  DISP=SHR,DSN=<Name.of.unpacked.file>(DBV7BDB2)
//SYSLIB   DD  DSN=DSNxxx.SDSNLOAD,DISP=SHR          DB2 Support
//         DD  DSN=IMSxxx.SDFSRESL,DISP=SHR          IMS Support
//         DD  DSN=CSQxxx.SCSQLOAD,DISP=SHR          MQS Support
//         DD  DSN=CICSTSxx.CICS.SDFHEXCI,DISP=SHR   CICS Support
//SYSPRINT DD  SYSOUT=*
```

Fig. 3: Linkage Editor DB2 Version 7

> Copy or rename member DBV7DBRM to member QPACBDB2.

> Bind QPACBDB2, e.g.

```
//BIND     EXEC PGM=IKJEFT01,REGION=2048K,COND=(4,LT),
//              PARM='DYNMNBR=25'
//STEPLIB  DD DISP=SHR,DSN=DSN710.SDSNLOAD
//SYSTSPRT DD SYSOUT=*
//SYSTERM  DD SYSOUT=*
//DBRMLIB  DD DISP=SHR,DSN=<Name.of.unpacked.File>
//SYSTSIN  DD *
DSN SYSTEM(<db2id>)
BIND  PACKAGE(<collection-id>)    MEMBER(QPACBDB2)       -
      ISOLATION(CS)       VALIDATE(BIND)         -
      DEGREE(ANY)         DBPROTOCOL(DRDA)       -
      CURRENTDATA(NO)     OWNER(QPAC)            -
      EXPLAIN(NO)         FLAG(I)                -
      DYNAMICRULES(RUN)                          -
      ACTION(REPLACE)
BIND  PLAN(<planname>)    PKLIST(*.<collection-id>.*)    -
      OWNER(QPAC)         ACTION(REPLACE)
/*
```

Fig. 4: BIND DB2 Version 7

DB2 version 8:

> For DB2 version 8 support the corresponding DB2 version 8 QPAC modules must be installed.
Therefore process member DBV8BDB2 by the linkage editor.

```
//LKED     EXEC PGM=HEWLH096,PARM=(XREF,LET,LIST,MAP),
//              REGION=6144K
//SYSUT1   DD  UNIT=SYSALLDA,SPACE=(1024,(200,20))
//SYSLMOD  DD  DISP=SHR,DSN=<QPAC.linklib>
//SYSLIN   DD  DISP=SHR,DSN=<Name.of.unpacked.file>(DBV8BDB2)
//SYSLIB   DD  DSN=DSNxxx.SDSNLOAD,DISP=SHR          DB2 Support
//         DD  DSN=IMSxxx.SDFSRESL,DISP=SHR          IMS Support
//         DD  DSN=CSQxxx.SCSQLOAD,DISP=SHR          MQS Support
//         DD  DSN=CICSTSxx.CICS.SDFHEXCI,DISP=SHR   CICS Support
//SYSPRINT DD  SYSOUT=*
```

Fig. 5: Linkage Editor DB2 Version 8

> Copy or rename member DBV8DBRM as member QPACBDB2.

> If needed delete or re-rename QPACBDB2 (DBRM V7).

> Bind QPACBDB2 e.g.

```
//BIND     EXEC PGM=IKJEFT01,REGION=2048K,COND=(4,LT),
//              PARM='DYNMNBR=25'
//STEPLIB  DD DISP=SHR,DSN=DSN810.SDSNLOAD
//SYSTSPRT DD SYSOUT=*
//SYSTERM  DD SYSOUT=*
//DBRMLIB  DD DISP=SHR,DSN=<Name.of.unpacked.File>
//SYSTSIN  DD *
 DSN SYSTEM(<db2id>)
BIND  PACKAGE(<collection-id>)    MEMBER(QPACBDB2)       -
       ISOLATION(CS)        VALIDATE(BIND)          -
       DEGREE(ANY)          DBPROTOCOL(DRDA)        -
       CURRENTDATA(NO)      OWNER(QPAC)             -
       EXPLAIN(NO)          FLAG(I)                 -
       DYNAMICRULES(RUN)                            -
       ACTION(REPLACE)
 BIND  PLAN(<planname>)     PKLIST(*.<collection-id>.*)    -
       OWNER(QPAC)          ACTION(REPLACE)
```

Fig. 6: BIND DB2 Version 8

QPAC-Online Data Sets

QPAC-Online requires its own data sets to operate. These are all VSAM KSDS data sets, whose purpose is explained in the following sections.

QPACOMF

This data set (Online Main File) acts as a central information receiver. System data, user profile information, resource profile information etc. are stored in this data set. It must be defined before the first Sign On can be processed.

```
// EXEC PGM=IDCAMS
   DELETE (QPAC.ONLINE.MF) CLUSTER PURGE   -
          CAT (USER.CATALOG)
   DEFINE CLUSTER                         ( -
           NAME (QPAC.ONLINE.MF)            -
           CYL|BLK (pppppp ssssss)          -
           KEYS (23 0)                      -
           RECSZ (128 256)                  -
           INDEXED                          -
           VOL (vvvvvv)                     -
           SHR (4  )                      ) -
          DATA                            ( -
           NAME (QPAC.ONLINE.MF.DATA)       -
           CISZ (4096)                    ) -
          INDEX                           ( -
           NAME (QPAC.ONLINE.MF.INDEX)      -
           CISZ (2048)                    ) -
          CAT (USER.CATALOG)
/*
```

Fig. 7: QPACOMF cluster definition

QPACOLB

This data set (QPACOLx Library) is for application development. QPAC-Online maps, programs etc. are stored in this data set. It is primarily used by the QPAC-Online full screen editor. The last letter of the name (DDname, DLBL), in this case “B”, is the library identification. Depending on organisational requirements, several libraries can be defined in alphabetical order (e.g. QPACOLC, QPACOLD).
The libraries used are assigned in the user profiles (see later in this manual).

```
// EXEC PGM=IDCAMS
   DELETE (QPAC.ONLINE.B) CLUSTER PURGE    -
          CAT (USER.CATALOG)
   DEFINE CLUSTER                         ( -
           NAME (QPAC.ONLINE.B)             -
           CYL|BLK (pppppp ssssss)          -
           KEYS (22 0)                      -
           RECSZ (1000 4020)                -
           INDEXED                          -
           VOL (vvvvvv)                     -
           SHR (4  )                      ) -
          DATA                            ( -
           NAME (QPAC.ONLINE.B.DATA)        -
           CISZ (8192)                    ) -
          INDEX                           ( -
           NAME (QPAC.ONLINE.B.INDEX)       -
           CISZ (4096)                    ) -
          CAT (USER.CATALOG)
/*
```

Fig. 8: QPACOLB cluster definition

Ensure that the empty status is removed from the VSAM files after definition.
You can do this by - for example - loading a dummy record using QPAC-Batch and then immediately deleting it.

```
//JOBNAME   JOB     .............
//          EXEC    PGM=QPAC
//STEPLIB   DD      DSN=...
//QPACLIST  DD      SYSOUT=*
//QPACOMF   DD      DSN=QPAC.ONLINE.MF,DISP=SHR
[ //QPACOLB DD     DSN=QPAC.ONLINE.B,DISP=SHR ]
//QPACIN    DD      *
OPF=*QPACOMF,VSAM
END
//          EXEC    PGM=QPAC
//STEPLIB   DD      DSN=...
//QPACLIST  DD      SYSOUT=*
//QPACOMF   DD      DSN=QPAC.ONLINE.MF,DISP=SHR
//QPACIN    DD      *
UPF=*QPACOMF,VSAM
PUTD GOBACK
END
```

Fig. 9: Reset VSAM empty status

QPAC-Online CICS Table Entries

RDO Definitions (DFHFCT, DFHPCT, DFHPPT)

```
// EXEC DFHCSDUP,SIZE=600K     UPDATE CICS CSD VSAM FILE
    VERIFY
     DELETE GROUP(QPAC)
     DEFINE PROFILE(QPACPROF) G(QPAC) UCTRAN(NO)
     DEFINE FILE(QPACOMF) G(QPAC) DSNAME(QPACOMF.TST1) READ(YES)
                                  DATABUFFERS(6)       ADD(YES)
                                  INDEXBUFFERS(5)      DELETE(YES)
                                  STRINGS(5)           BROWSE(YES)
                                  LSRPOOLID(NONE)      UPDATE(YES)
     DEFINE FILE(QPACOLB) G(QPAC) DSNAME(QPACOLB.TST1) READ(YES)
                                  DATABUFFERS(6)       ADD(YES)
                                  INDEXBUFFERS(5)      DELETE(YES)
                                  STRINGS(5)           BROWSE(YES)
                                  LSRPOOLID(NONE)      UPDATE(YES)
*    DEFINE FILE(QPACOLC) G(QPAC) DSNAME(QPACOLB.TST1) READ(YES)
*                                 DATABUFFERS(6)       ADD(YES)
*                                 INDEXBUFFERS(5)      DELETE(YES)
*                                 STRINGS(5)           BROWSE(YES)
*                                 LSRPOOLID(NONE)      UPDATE(YES)
     DEFINE MAPSET (QPACOME)  G(QPAC)
     DEFINE MAPSET (QPACOM0)  G(QPAC)
     DEFINE MAPSET (QPACOM1)  G(QPAC)
     DEFINE PROGRAM(QPACONLA) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACONLB) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACONLD) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACONLG) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACONLI) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACONLJ) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACONLM) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACONLR) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACONLT) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACONL0) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACONL1) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACOSON) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACO100) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACO200) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACO300) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACO400) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACO410) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACO420) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACO430) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACO500) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACO600) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACO610) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACO700) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACO710) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACO720) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACO800) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACO810) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACO820) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACO860) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACO900) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACOSM0) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACOSP0) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACOKS0) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACOES0) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACORR0) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACOTS0) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACOTD0) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACOPH0) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACOPH1) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACOPH2) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACOPH3) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACOPH4) G(QPAC) LANGUAGE(ASSEMBLER)
```

```
     DEFINE PROGRAM(QPACOPH5) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACOPH6) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACOPH8) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACOPHA) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACOPHC) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACOPHE) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACOPHI) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACOPHR) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACOPHX) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACOF11) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACOF12) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACOF28) G(QPAC) LANGUAGE(ASSEMBLER)
*-------------------------------------------------------------
* FOLLOWING PROGRAMS FOR DL/I SUPPORT
*-------------------------------------------------------------
     DEFINE PROGRAM(QPACODL0) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACODLI) G(QPAC) LANGUAGE(ASSEMBLER)
*-------------------------------------------------------------
* FOLLOWING PROGRAMS FOR DB2 SUPPORT
*-------------------------------------------------------------
     DEFINE PROGRAM(QPACOQL0) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACOPHQ) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACODB0) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACODB2) G(QPAC) LANGUAGE(ASSEMBLER)
*-------------------------------------------------------------
* FOLLOWING PROGRAMS FOR MQSERIES SUPPORT
*-------------------------------------------------------------
     DEFINE PROGRAM(QPACOMQ0) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACOMQE) G(QPAC) LANGUAGE(ASSEMBLER)
*-------------------------------------------------------------
* FOLLOWING PROGRAMS FOR WEB SUPPORT
*-------------------------------------------------------------
     DEFINE PROGRAM(QPACOWB0) G(QPAC) LANGUAGE(ASSEMBLER)
     DEFINE PROGRAM(QPACOWBE) G(QPAC) LANGUAGE(ASSEMBLER)
```

Fig. 10: RDO definitions PROGRAM, MAPSET

```
*-------------------------------------------------------------
* TRANSACTIONS GENERAL
*-------------------------------------------------------------
     DEFINE TRANSACTION(QPAC) G(QPAC) PROGRAM(QPACONLA) TWASIZE(0)
                                      PROFILE(QPACPROF) PRIORITY(40)
     DEFINE TRANSACTION(QPBC) G(QPAC) PROGRAM(QPACONLB) TWASIZE(0)
                                      PROFILE(QPACPROF) PRIORITY(200)
     DEFINE TRANSACTION(QPJC) G(QPAC) PROGRAM(QPACONLJ) TWASIZE(0)
                                      PROFILE(QPACPROF) PRIORITY(40)
     DEFINE TRANSACTION(QPMC) G(QPAC) PROGRAM(QPACONLM) TWASIZE(0)
                                      PROFILE(QPACPROF) PRIORITY(40)
     DEFINE TRANSACTION(QPTC) G(QPAC) PROGRAM(QPACONLT) TWASIZE(0)
                                      PROFILE(QPACPROF) PRIORITY(40)
     DEFINE TRANSACTION(QP0C) G(QPAC) PROGRAM(QPACONL0) TWASIZE(0)
                                      PROFILE(QPACPROF) PRIORITY(40)
     DEFINE TRANSACTION(QP1C) G(QPAC) PROGRAM(QPACONL1) TWASIZE(0)
                                      PROFILE(QPACPROF) PRIORITY(40)
*-------------------------------------------------------------
* FOLLOWING TRANSACTIONS AS MODELS FOR USER DEFINED TRANSACTIONS
* ------------------------------------------------------------
*  DEFINE TRANSACTION(....) G(QPAC) PROGRAM(QPACONLA) TWASIZE(0)
*                                   PROFILE(QPACPROF) PRIORITY(40)
*  DEFINE TRANSACTION(....) G(QPAC) PROGRAM(QPACONL1) TWASIZE(0)
*                                   PROFILE(QPACPROF) PRIORITY(40)
*-------------------------------------------------------------
* FOLLOWING FOR EXCI SUPPORT
*-------------------------------------------------------------
   DEFINE TRANSACTION(QPEC) G(QPAC) PROGRAM(DFHMIRS)
                                    PROFILE(QPACPROF)
                                    SPURGE(YES) TPURGE(YES)
                                    CONFDATA(YES)
   DEFINE CONNECTION(QPCS) G(QPAC) ACCESSMETHOD(IRC)
```

```
                                        PROTOCOL(EXCI)
                                        NETNAME(QPACEXCI)
                                        CONNTYPE(SPECIFIC)
                                        SINGLESESS(NO)
                                        INSERVICE(YES)
                                        ATTACHSEC(LOCAL)
   DEFINE SESSIONS(QPCS) G(QPAC) CONNECTION(QPCS)
                                        PROTOCOL(EXCI)
                                        RECEIVECOUNT(5)
                                        RECEIVEPFX(QS)
                                        RECEIVESIZE(4096)
                                        SENDSIZE(4096)
                                        IOAREALEN(4096,4096)
                                        INSERVICE(YES)
```

Fig. 11: RDO definitions TRANSACTION, CONNECTION, SESSION

```
*-------------------------------------------------------------------
* TRANSIENT DATA QUEUES
*-------------------------------------------------------------------
*  DEFINE TDQUEUE  (QPTC) G(QPAC) INDIRECTNAME(XXXX) TYPE(INDIRECT
*  DEFINE TDQUEUE  (XXXX) G(QPAC) TRANSID(QPTC) TYPE(INTRA)
   ADD  G(QPAC) LIST(XXXLIST)
/*
```

Fig. 12: RDO definitions TDQUEUE

WEB Support Feature

If required the corresponding TEMPLATES have to be defined in CICS RDO, e.g.

```
DEFINE DOCTEMPLATE (templname) G(QPAC)
       TEMPLATENAME(templname)
       MEMBERNAME  (templname)

DEFINE ....
```

Fig. 13: TEMPLATE entries

DFHDCT

```
QPTC     DFHDCT TYPE=INDIRECT,DESTID=QPTC,                          X
                INDDEST=xxxx
xxxx     DFHDCT TYPE=INTRA,DESTID=xxxx,TRIGLEV=1,                   X
                TRANSID=QPTC,DESTFAC=TERMINAL
```

Fig. 14: DFHDCT table entries

DFHPLT (Post Initialization PLTPI)

```
QPACONLI DFHPLT TYPE=ENTRY,PROGRAM=QPACONLI            *)
```

*) optional: automatically activates the QPBC background transaction

Fig. 15: DFHPLTPI table entries

DFHPLT (Shutdown PLTSD, 1st quiesce stage)

```
QPACONLB DFHPLT TYPE=ENTRY,PROGRAM=QPACONLB
```

Fig. 16: DFHPLTSD table entries

DFHSIT

verify inclusion of:

```
          SPOOL=YES,
```

Fig. 17: DFHSIT table entries

QPAC-Online First Time Sign On Procedure

This **first time sign on** to QPAC-Online causes the initialization of the system administration record on the QPACOMF data set. If this procedure is not successful it must be restarted, starting with a DELETE/DEFINE CLUSTER for the QPACOMF data set.
The first time sign on also requires the definition of a system master password and an administrator profile.

You are now asked to sign on with the user id **INSTALLATION** and to enter a system password. It is important to remember this password. It can be used when no valid user password is available.

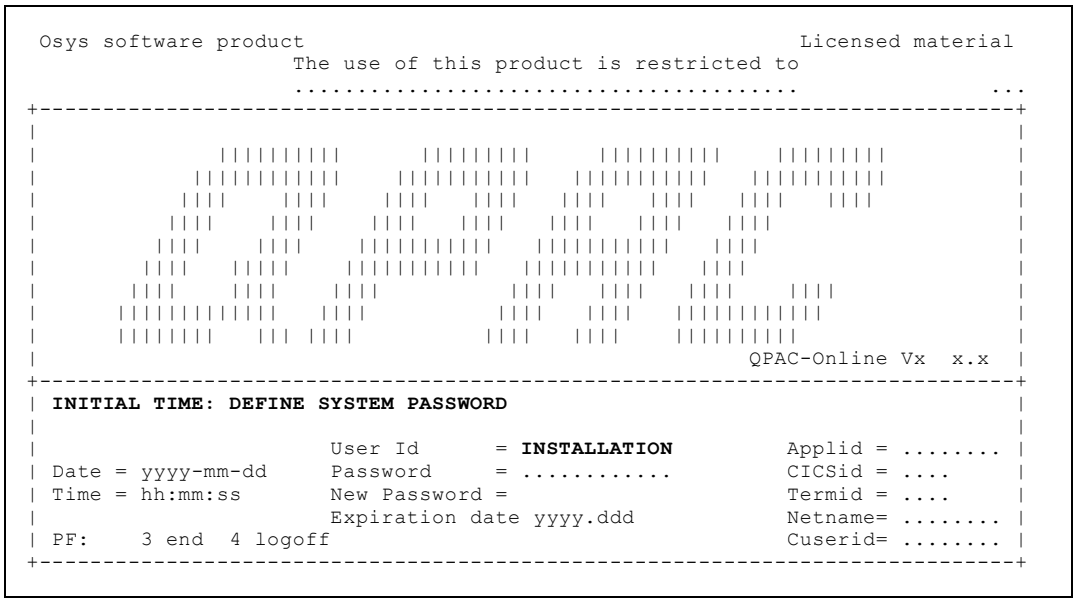


Fig. 18: The QPAC-Online first time sign on

The screen panel for defining user profiles then appears.

Define an administrator user id with all the required authorities. Please consult the chapter “QPAC-Online System Administration”, section “User Profile Definition” for further explanation of this panel.

After successful definition of the administrator user press the PF3 key to **SIGN OFF**.

The QPAC-Online system is now ready for use. Consult the chapter “QPAC-Online System Administration” for explanation of system, user and resources profile definitions.

QPAC-Online Sample Tape Installation

For test versions and first time installations an optional second tape will be supplied, containing a few simple examples with the aid of which the first exercises can be carried out. The tape contains the sublibrary DEMO, and was produced with the `BACKUP` function of the maintenance utility QPACUTIL.

Use one of the following jobs to restore the sample tape:

```
//Jobname   JOB    SAMPLE
//          EXEC   PGM=QPACUTIL
//QPACRST   DD     DSN=QPACOLB.DEMO,
//                 DCB=(RECFM=VB,LRECL=8000,VOL=SER=QPDEMO,
//                 BLKSIZE=16000),UNIT=TAPE,DISP=OLD,VOL=SER=QPDEMO
//QPACOLB   DD     DSN=........,DISP=SHR
//QPACLIST  DD     SYSOUT=*
//QPACIN    DD     *
RESTORE *.*.*   REPLACE=YES
/*
//
```

Fig. 19: Load sample tape

After the restoration of the sample tape define a user with the sublibrary DEMO assigned to it. You can now sign on with the demo user.

Chapter 2. QPAC-Online System Administration

Overview

The definition of individual priviliges and the system layout is the responsibility of the system administrator. Here we differentiate between four different types of definitions:

1. Definitions concerning the whole system
2. Definitions concerning the system user
3. Definitions concerning resources, data file security
4. Definitions concerning programs, shared use

A further responsibility of the system administrator is the question of implementation with respect to the operating system, for example:

1. The definition of the job control skeletons for the assembling of QPAC-Online map definitions in CICS BMS macro format.

System Profile Definition

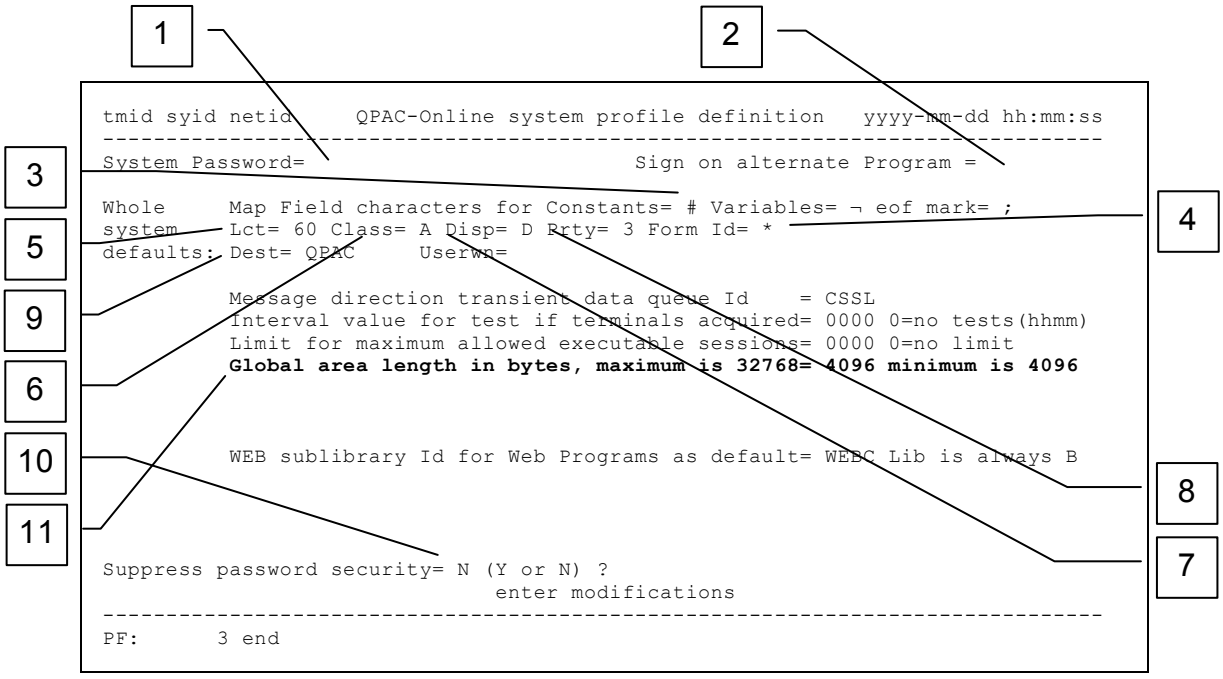


Fig. 20: The system profile screen

This screen is used for definitions which apply to the system in general:

1 **System password**
With this password, a defined user can access the administrator menu at any time. Knowledge of the password should therefore be limited.

The system password, which is defined at installation time, can be changed at any time by the authorised user.
The first character input for the system password is not a part of the password itself. If a `D` (display) is input, the password is visible, otherwise it is not.

2 **Sign on alternate program**
The official sign on process can be overridden by a user written program. This program will be jumped to with LINK before the sign on screen appears. The user id will be expected as return data in the COMMAREA (and perhaps a password), with which the user profile can be accessed.

```
EXEC CICS LINK
   PROGRAM  (USERPROG)
   COMMAREA (USERAREA)
   LENGTH   (32)
```

Fig. 21: Calling a sign on alternate program

The user id must be returned in the first 16 bytes of the COMMAREA. If a password is returned in the second 16 bytes, and the password check is not suppressed, it will be compared with that of the user profile. If the first character of the password part of the COMMAREA is blank, it is assumed that the password is not to be checked.

Another possibility is the specification of the predefined program **QPACOSON** in field number 2.
This program evaluates the CICS user id. If it corresponds to a valid QPAC-Online user id the QPAC-Online sign on procedure will be bypassed and the main menu screen displayed.

3 **Map field characters**
Map field constants and characters are symbols that can be used for defining fields when developing maps. The following defaults are used:

#	for constant fields
¬	for variable fields (^ for swiss keyboard)
;	for field end marks (FIELD MARK key)

Fig. 22: Map field characters overview

The default symbols are used in all examples in the QPAC-Online literature.

4 **Forms identification**
If when using z/OS JES, a form identification is to be assigned, it can be entered here.
An asterisk means a blank forms id.

5 **Line counter**
The standard page size for output listings can be defined here.

6 **Output list class**
The standard output class for output listings can be defined here.

7 **Disposition**
The standard disposition for output lisitings can be defined here.
D = Nohold, H = Hold.

8 **Priority**
The standard priority from 1 to 9 for output listings can be defined here.

9 **Destination**
The destination as a remote id can be defined here.
A definition * or LOCAL means system line printer as destination.

10 **Suppress password security**
If the user does not require any password control, this can be defined here. Thereby the password is redundant, which is sometimes a required feature in simple installations.

11 **Global area**
This is an area within an application that exists also after any QLINK or QXCTL. It has a minimum size of 4K and a maximum size of 32K.

User Profile Definition

With this screen, the users wishing to use QPAC-Online are defined. Furthermore, each user is given a password. The individual privileges are defined separately, generally with the input of either `Y` (yes) or `N` (no).

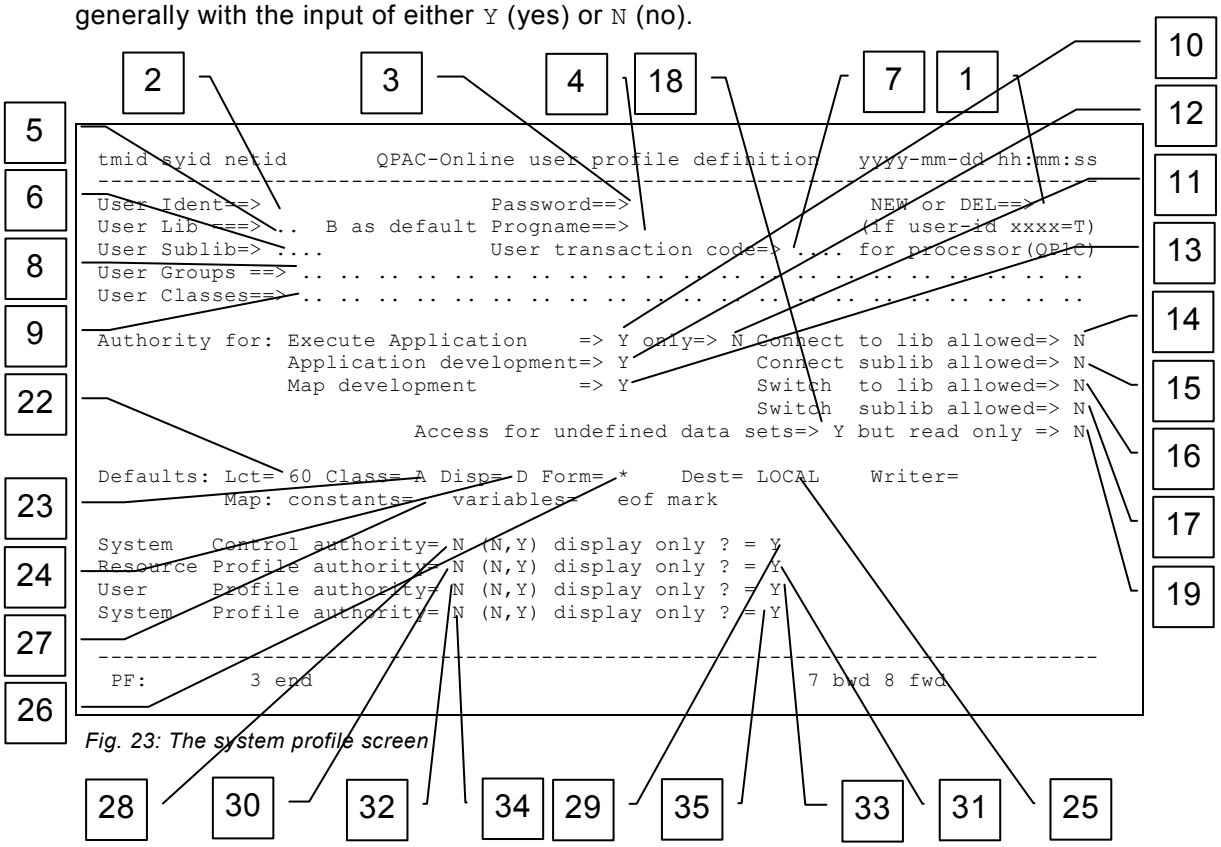


Fig. 23: The system profile screen

1 The `NEW` definition plus the other input defines a new user.
The `DEL` definition plus the user ident input, deletes an existing user.

The input of just a user ident in field 2 enables the modification of an existing user. In this case field 1 is left blank.

2 **User Ident**
Is the name of the privileged user, 1-16 characters long. The user profile record is entered under this name in the QPACOMF dataset.

Instead of a user id a **transaction code** can be entered according to the following format: `xxxx=TRANSACTION` (first two bytes of transaction code must **not** be `QP`). If such a transaction code is being defined a **QPAC-Online program name** is to be entered in field 4 instead of a password in field 3.
The transaction code `xxxx` must also be defined in the CICS RDO with the same parameters as the transaction QPAC.
With this form of definition it is possible to assign an individual transaction code to a QPAC-Online application. This QPAC-Online application can so directly be started from outside the QPAC-Online environment by entering the corresponding `xxxx` transaction code, bypassing the sign on procedure.

3 **Password**
Is the password that must be used at sign on by the user as long as password checking has not been suppressed.

4 **Progname**
If instead of a user id a transaction code has been filles in field 2, a QPAC-Online program name must be filles in field 4 instead of a password (field 3). This application is automatically started when the specified transaction code is entered from outside the QPAC-Online environment.

5 **User Lib**
This definition assigns a library to the user. Two letters may be specified in z/OS, which will be the last characters of the file name QPACOL*xx*. This library must also be defined in the CICS RDO and be allocated as a VSAM cluster.
If omitted, the default is `B` (QPACOL`B`).

6 **User Sublib**
Is a sublibrary prefix for the user. This sublibrary prefix separates the user from other users within the physical library. The sublibrary is a virtual separation of the library.
Access is only possible to programs within this sublibrary, as long as the user is not explicitly authorised to change to other libraries (`SWITCH` command).
The definition can be a four character alphanumeric term. The first character must be alphabetic. If omitted, the default is `BASE`.
Purely numeric sublibrary definitions are reserved for internal use.

7 **User Transaction code**
QPAC-Online programs are executed under the transaction code defined here. Through the individual assignment of transaction codes, it is possible to utilise the CICS security concept, in that the security keys are distributed to groups or users, which can here be applied to QPAC-Online programs. It is however not compulsory to define an individual transaction code. Such a transaction code must also be defined in the CICS RDO.
If no transaction code is defined, **QP1C** is used as the default.

8 **User Groups**
These are two digit numbers from `01` - `99`, which are used for classifying programs in a sublibrary into groups. The individual programs can be allocated a group number in menu screen “Application Development”. On this screen the user sees only those programs with the same group number.
The group number `00` can not be defined, and is taken as **generally valid**.

9 **User Classes**
These are two digit numbers from `01` - `99`, which serve as classification numbers within a group. This classification is effective in menu “Execute Application”, where only those programs whose group and class are defined in the user profile appear. Thereby, for example, main and sub-programs can be separated for the selection menu.
Class `00` does **not** appear in the directory.

Authorities
This is the authorisation for various applications:

10 **Execute Application**
Allows access to menu 9 “Execute Application”.

11 **Execute Application only**
Allows the user to work only with menu 9 “Execute Application”. In this case, the user is placed in this menu immediately after the sign on. This would be the case for pure users, who do not undertake any development work.

12 **Application Development**
Allows access to menu 8 “Application Development”. This function is a full screen editor, with whose help QPAC-Online programs or applications can be developed. These programs will be stored in either the dataset QPACOLB or QPACOL*x*.

13 **Map Development**
Allows access to menu 7 “Map Development”. This function is a full screen map editor, with whose help maps can be developed. These maps are stored either in the dataset QPACOLB or QPACOL*x*.

14 **Connect Lib(rary) Command**
Privilege allows a connection to a library (QPACOLx) to which members may be moved (m) or copied (c) from within the application or map development menu.

15 **Connect Sublib Command**
Privilege allows a connection to a sublibrary to which members may be moved (m) or copied (c) from within the application or map development menu.

16 **Switch Lib(rary) Command**
Privilege allows a change of the Library within the application or map development menu.

17 **Switch Sublib Command**
Privilege allows a change of the sublibrary within the application or map development menu.

18 **Dataset Access**
Defines whether files which are not explicitly listed within menu 3 “Resource Definition” may be accessed.
`Y` (yes) states that undefined datasets can generally be accessed.
`N` (no) states that undefined datasets can generally not be accessed.

19 **Read only Dataset Access**
Only “read” access is granted to undefined datasets.

20 **reserved**

21 **reserved**

User defaults:
For missing definitions in the user profile (blank), those from the system profile are taken.

22 **Line Counter**
The standard page size for output listings can be defined here.

23 **Output List Class**
The standard output list class for output listings can be defined here.

24 **Disposition**
The standard disposition for output listings can be defined here.
`D` = Nohold, `H` = Hold.

25 **Destination**
The destination can be defined here. A definition of `*` or `LOCAL` means system lineprinter as destination.

26 **Forms Identification**
If when using z/OS JES, a form identification is to be assigned, it can be

entered here.
An asterisk means a blank forms id.

27 **Map field characters**
Map field constants and characters are symbols that can be used for defining fields when developing maps. The following defaults are used:

#	for constant fields
¬	for variable fields (^ for swiss keyboard)
;	for field end marks (`FIELD MARK` key)

Fig. 24: Map field characters overview

The default symbols are used in all examples in the QPAC-Online literature.

28 **System Control Authority**
This allows logged on users, users in the execution phase, programs which are dynamically assembled and thereby resident, to be shown.

29 **System Control Authority display only**
Allows activity to be viewed, but not changed.

30 **Resources Profile Authority**
This is the authority for the management of dataset access privileges.

31 **Resources Profile Authority display only**
Allows profiles to be viewed, but not changed.

32 **User Profile Authority**
This is the authority for the management of user profiles.

33 **User Profile Authority display only**
Allows user profiles (but not the passwords) to be viewed, but not changed.

34 **System Profile Authority**
This is the authority for the management of the system profile.

35 **System Profile Authority display only**
Allows the system profile (but not the system password) to be viewed, but not changed.

Resources Profile Definition

With this screen, the data files which cannot be accessed, or only selectively or restrictively, are defined.
In this profile CICS datasets can be defined, to which the access privileges are not the same for all users.

A dataset, whose profile record is not available to this screen is, depending on the user profile definition, either generally accessable, or generally not accessable.

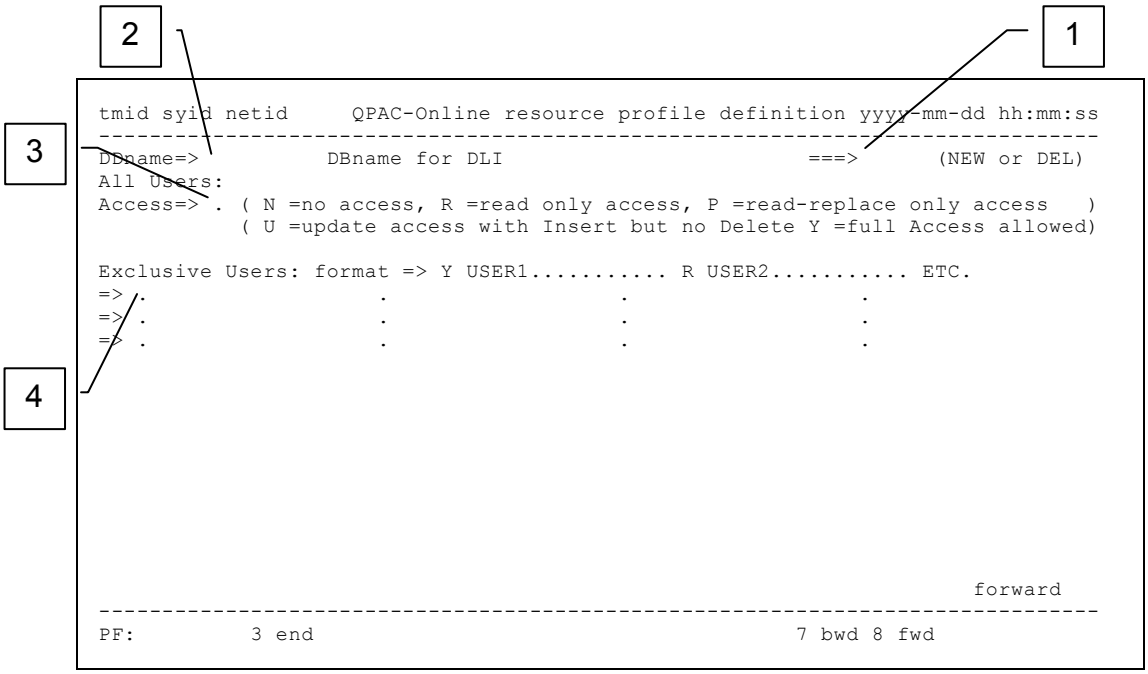


Fig. 25: The resources profile screen

1 The `NEW` definition plus the other input defines a new dataset profile.
The `DEL` definition plus the DDname input, deletes an existing dataset profile.

The input of just a DDnaame in field 2 enables the modification of an existing dataset. In this case field 1 is left blank.

2 **DDname**
The dataset is identified by its DDname.

3 **General Privileges**
`N` = Generally no access privilege
`Y` = Universal access privilege
`R` = Read only
`P` = Read and write back (replace)
`U` = Read, write back and insert (update)

4 **Exclusive Users**
Here, users whose privileges differ from those stated under the general access definitions, can be defined.
The privilege code and the user identification should be entered.

System Control Functions

With this menu, control functions can be called:

1. Which users are logged in.
2. Which users are in processing mode, i.e. executing programs (sessions).
3. Which programs are being used, i.e. assembled.

Furthermore, the programs which should generally be resident, can be defined on this screen. The programs stipulated here will automatically be assembled when the first QPAC-Online user signs on.

Logged On Users

When menu “System Control Functions” is called, the screen shows which users are logged on.

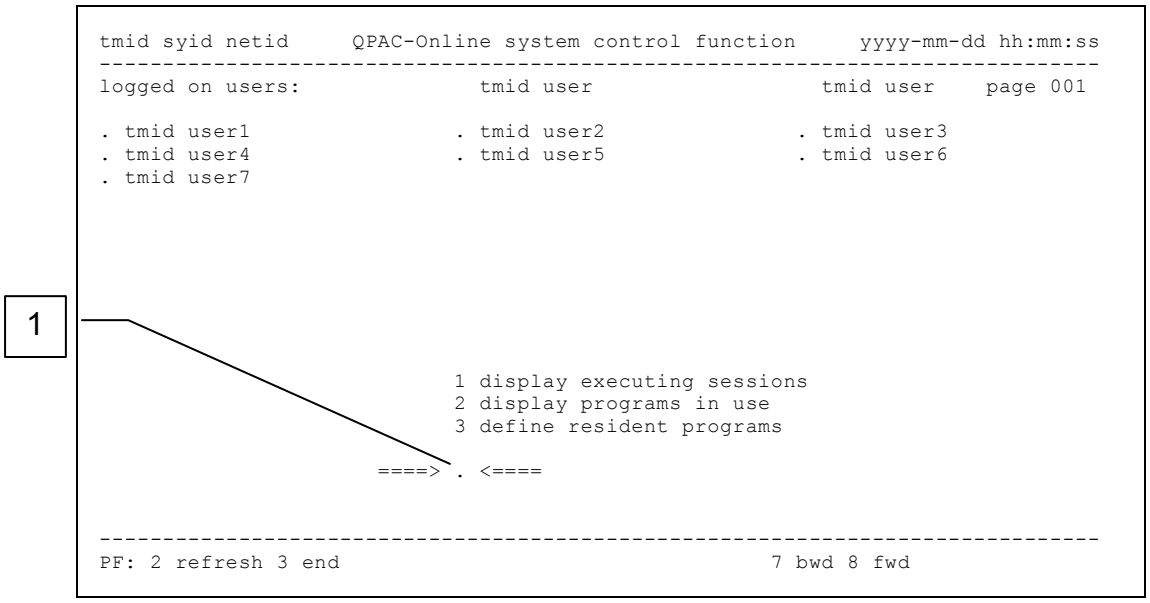


Fig. 26: The resources profile screen

1 By entering a sub-selection code, further control screens can be called which are described on the following pages.

Executing Sessions

The sub-selection code 1 calls the screen showing the users in processing mode at that time, i.e. executing programs (applications).

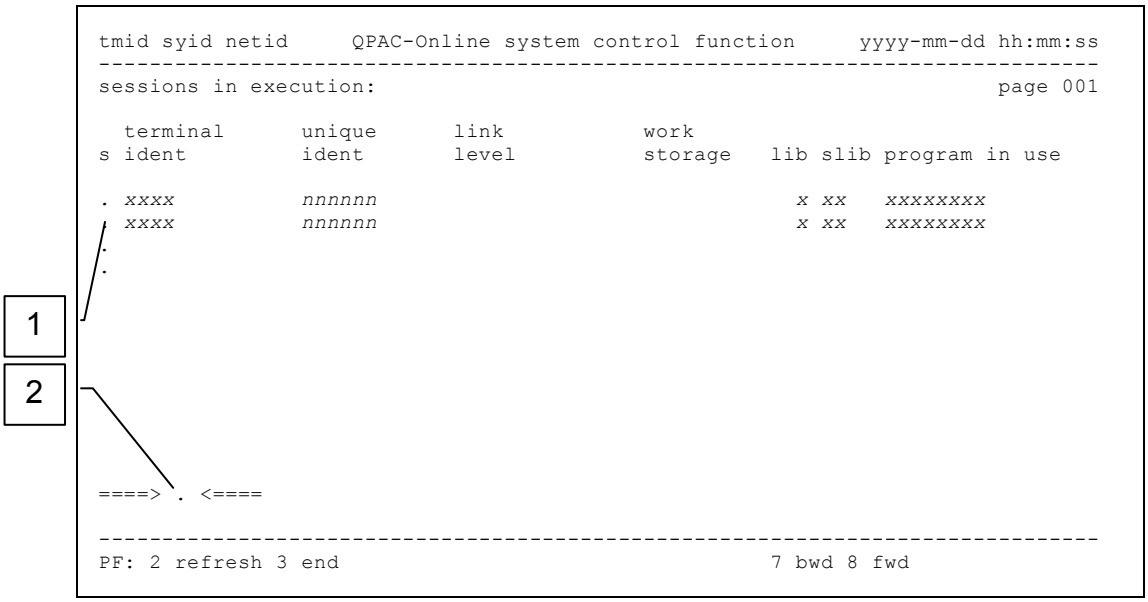


Fig. 27: The system control functions screen (executing sessions)

1 `F` = Force off

A session can be ended with the `F` command, as long as the user is authorised to do so. See the user profile.

2 By entering a sub-selection code, further control screens can be called:

`0` = “logged on users” base screen
`2` = “programs in use” screen
`3` = “resident programs” screen.

The `PF2` key refreshes the status of the screen.

Programs In Use

The sub-selection code 2 calls the screen showing the programs which are assembled and made available for execution by the background storage manager.

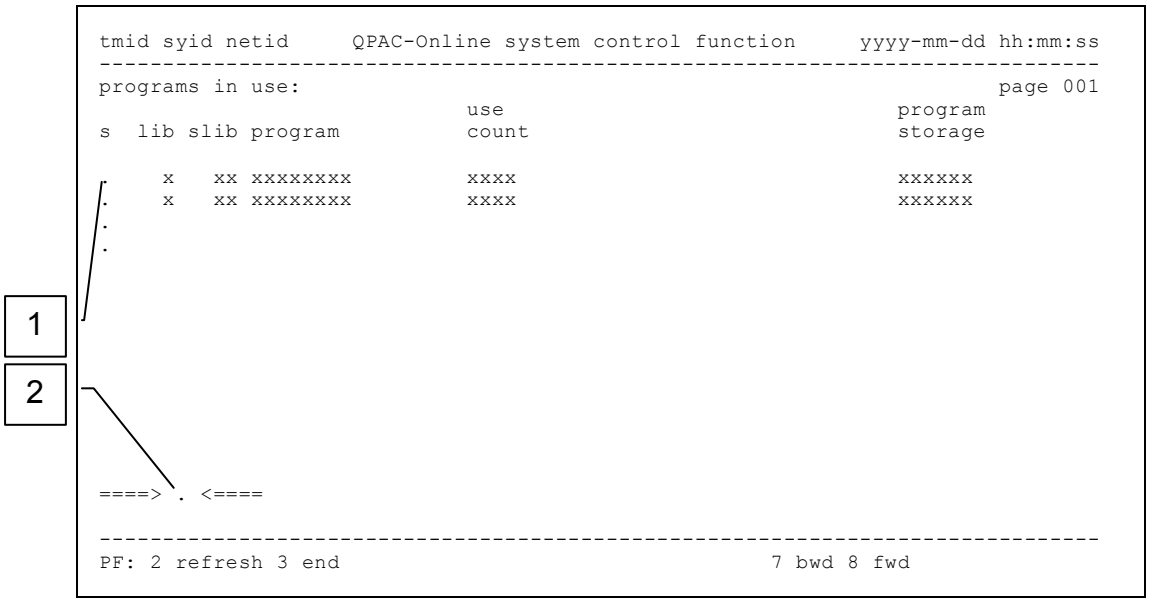


Fig. 28: The system control functions screen (programs in use)

1 **– reduce the use counter by 1**

The use counter can be reduced by 1 with the - command.
Thereby, the status of a resident program can be preserved.

+ increase the use counter by 1

The use counter can be increased by 1 with the + command.
Thereby, a program can be made resident.

2 By entering a sub-selection code, further control screens can be called:

`0` = “logged on users” base screen
`1` = “sessions in execution” screen
`3` = “resident programs” screen.

The `PF2` key refreshes the status of the screen.

Start Up Program Initialisation

The sub-selection code 3 calls the screen showing the programs which can be defined or managed, which are resident, and which will be assembled when QPAC-Online is started. The use counter 1 is assigned to these programs. They thereby remain resident, and are used in shared mode when a user executes one of these programs.

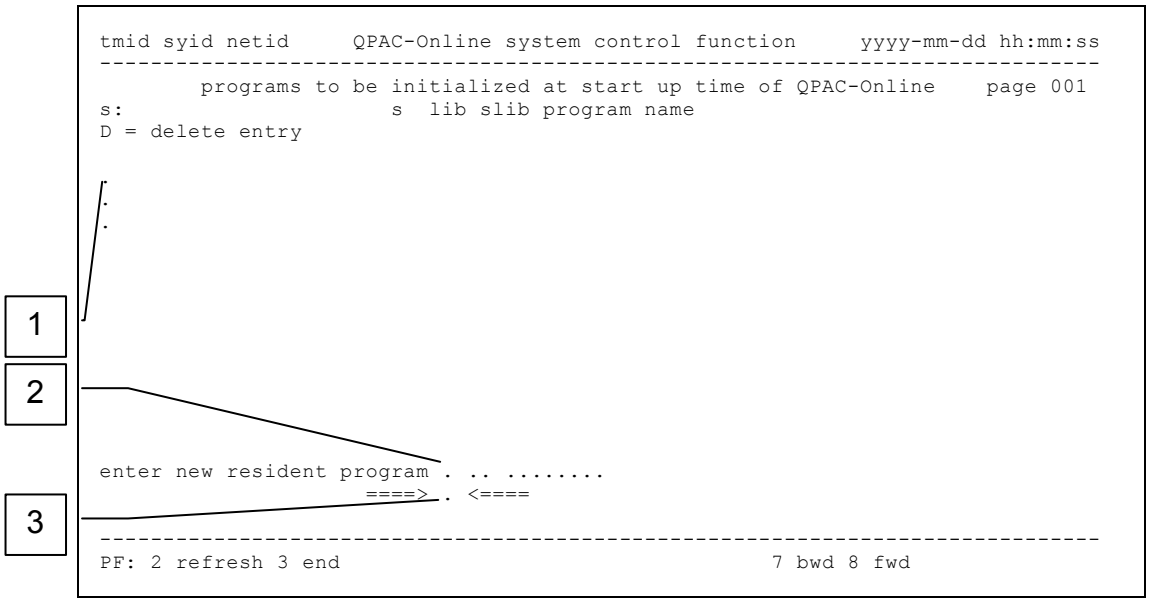


Fig. 29: The system control functions screen (resident programs)

1 `D` = **Delete an entry**
The command `D` (delete) deletes an existing entry from the table.

2 **New entries**
(programs) are defined in the given line.

3 By entering a sub-selection code, further control screens can be called:

`0` = “logged on users” base screen
`1` = “sessions in execution” screen
`2` = “programs in use” screen.

The `PF2` key refreshes the status of the screen.

Map Conversion to BMS Macro Format

In the “Map Development” menu, by a special order, QPAC-Online maps can be converted into BMS macro format, by generating and submitting a job. For the job to be generated the job control has to exist as a job skeleton in the sublibrary 0001 as well as under the mandatory name **JCLMASKMAP**. In the mentioned job skeleton some variables are included. They begin by a %-sign, to be efficient resp. necessary for control and for the completion.

%USER	1-4 positions for user id (default
%USER(n)	1-9 positions for user id explicitly defined
%OPID	3 positions for CICS operator id
%MAPSET	3-7 positions for map set name
%LANG	1 position for language **C**ob, **P**L/I, **A**sm

Fig. 30: Variables for BMS macro generation

```
>> *** JCL Mask for BMS z/OS                          >> 00 00 > JCLMASKMAP
                                                                        00000
1...5...10....5...20....5...30....5...40....5...50....5...60....5...70.X B.0001
*** Top of IWS ***                                                      =====
//BMS%USER JOB .............                                            =====
//ASM      EXEC PROC=DFHASMVS,ASSEM='SYSPARM(MAP)'                       =====
//SYSPUNCH DD   DSN=&TEMP,DCB=(RECFM=FB,BLKSIZE=2960),                   =====
//             SPACE=(2960,(10,10)),UNIT=SYSDA, DISP=(NEW,PASS)          =====
//SYSIN    DD   *                                                       =====
%DATAMAP                                                                =====
/*                                                                      =====
//LNK      EXEC PROC=DFHLNKVS,PARM='LIST,LET,XREF'                       =====
//SYSLIN   DD   DSN=&TEMP,DISP=(OLD,PASS)                                =====
//         DD   *                                                       =====
  NAME  %MAPSET(R)                                                      =====
/*                                                                      =====
//DSECT    EXEC PROC=DFHASMVS,ASSEM='SYSPARM(DSECT)'                     =====
//SYSPUNCH DD   DSN=USER.MAPLIB(%MAPSET),DISP=SHR                        =====
//SYSIN    DD   *                                                       =====
%DATADSECT                                                              =====
/*                                                                      =====
```

Fig. 31: Sample job skeleton for BMS z/OS

Call by VTAM USSTAB

QPAC-Online programs are able to be directly called by VTAM USS commands from within the USSTAB. The following definitions are necessary:

USSTAB Definitions

Sample:
```
USSCMD  CMD=command,REP=LOG,FORMAT=PL1
USSPARM PARM=APPLID,DEFAULT=cicsappl
USSPARM PARM=DATA,DEFAULT='QPACDATA=qtid'
```

command	is the command by which the user calls QPAC-Online
cicsappl	is the CICS applid where QPAC-Online is running
qtid	is the QPAC internal transaction code that is defined in the QPAC-Online user profile and is attached to the QPAC-Online program

Fig. 32: Sample USSTAB definitions

CICS SIT Definitions

'QPAC' must be defined as the "good morning transaction" in the CICS SIT. The QPAC sign-on program automatically starts the QPAC-Online user program based on the information provided by VTAM.
It is normally useful to run a separate mini CICS. But it is not absolutely necessary. With the VTAM definition `DEFAULT='QPACDATA=OFF'` an empty screen is presented instead of a program.

QPAC-Online User Profile

```
tmid syid netid       QPAC-Online user profile definition   yyyy-mm-dd hh:mm:ss
------------------------------------------------------------------------------
User Ident==> qtid=TRANSACTION Password==>                    NEW or DEL==>
User Lib ===> b   B as default Progname==> Program_name     (if user-id xxxx=T)
User Sublib=> slib             User transaction code=> .... for processor(QP1C)
User Groups ==> ..
..
```

Fig. 33: Sample user profile definitions for call by USSTAB

Chapter 3. QPAC-Online Batch Utility Functions

Overview

The planning and organisation of various system maintenance functions also comes under the responsibility of system administration. Included in this, for example, library security, the copying of individual programs and maps, the listing of library contents etc..

There are batch utilities and utility functions available to deal with this, whose usage is now described in detail.

The Batch Utility Program QPACUTIL

The batch utility **QPACUTIL** is a general help utility for the maintenance of the QPAC-Online library QPACOLB.

Usage

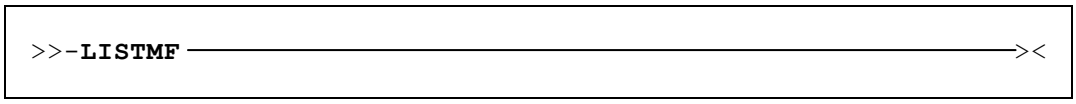
```
//Jobname  JOB ...
//Stepname EXEC  PGM=QPACUTIL [,PARM='command']
//QPACBUP  DD    DSN=.......,       for backup tape output
//               DCB=(RECFM=VB,LRECL=8000,....)
//QPACRST  DD    DSN=.......,       for restore tape input
//               DISP=OLD
//QPACOLB  DD    DSN=.......,       QPAC-Online library
//               DISP=SHR
//QPACLIST DD    SYSOUT=*
//QPACIN   DD    *
command
command
/*
//
```

Fig. 34: Sample JCL

Command Syntax and their Operands

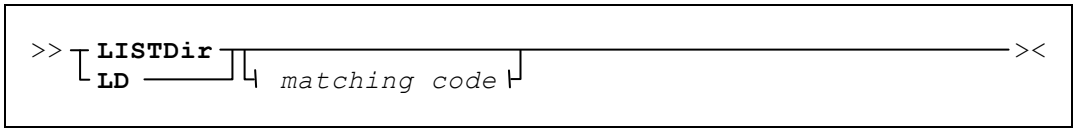
The LISTMF Instruction

List Main File



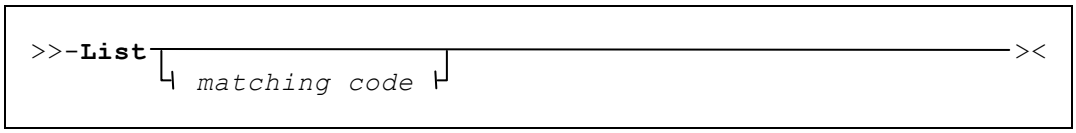
The LISTDIR Instruction

List Directory of Library



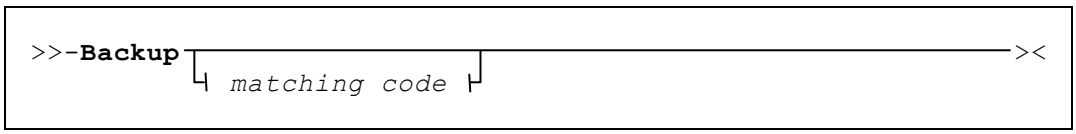
The LIST Instruction

List Member (program or map)



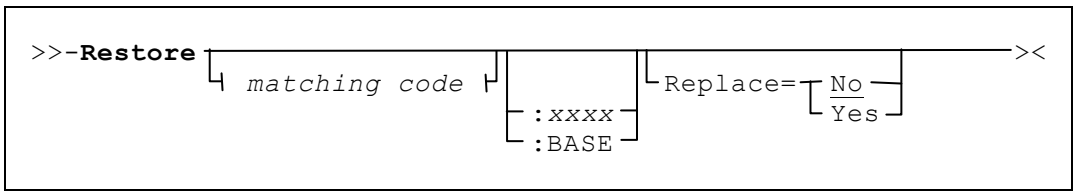
The BACKUP Instruction

Backup QPAC-Online Library



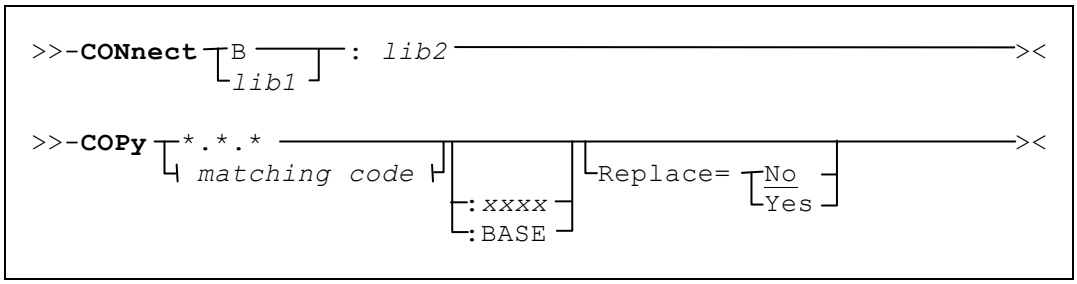
The RESTORE Instruction

Restore QPAC-Online Library



The CONNECT and COPY Instructions

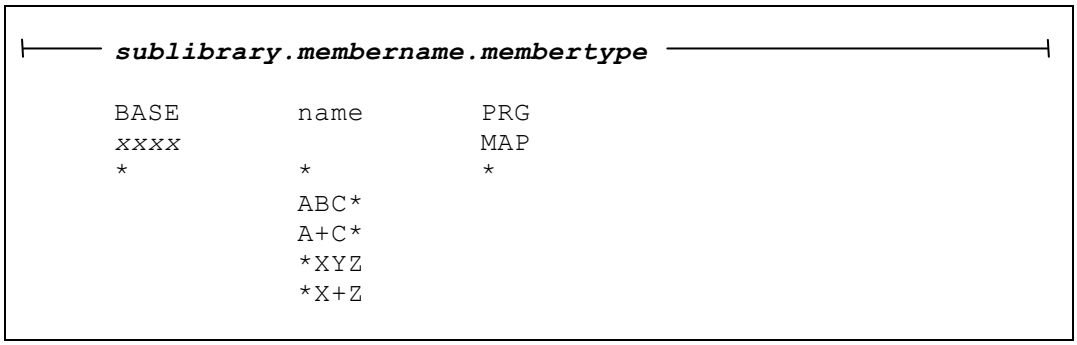
Connect two QPAC-Online Libraries and copy from/to library



lib1, *lib2* is the library identification letter (e.g. QPACOL*x*)

Syntax of the “Matching Code”

General Format

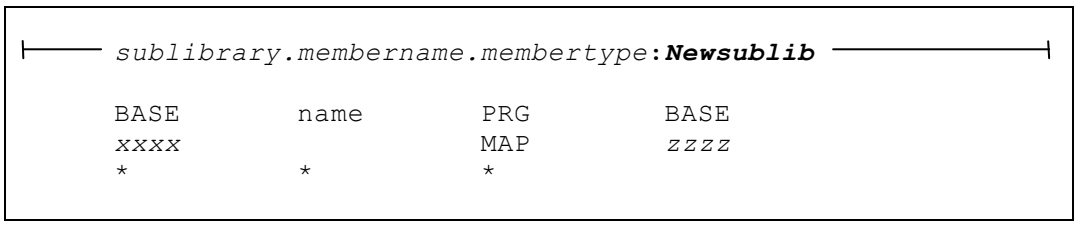


The three members of the “matching code” are defined concatenated by a full stop.

The member name can be selectively processed using the “pattern matching” principle with the aid of the two wildcards * and +, where + represents any character and * many characters.
The * can only appear at the beginning or at the end, and can only be defined once.
The + sign can be defined multiple times.

Special Format

The COPY and RESTORE commands allow the option of storing into another sublibrary. If this is required the new sublibrary must be appended to the “matching code” by a colon.



Chapter 4. DB2 Support Feature

Installation and Prerequisites under z/OS

1. Before the DB2 support feature can be used, the user must ensure that DB2 is installed as a product, and that the DB2 routines for batch processing are in the pre-determined libraries.

2. Following the catalog run of the QPAC product tape containing the DB2 support feature, under the official LINKAGE EDITOR conventions, it must be verified that the load modules **QPACBDB2 (QPAC-Batch)** and/or **QPACODB2 (QPAC-Online)** are correctly linked, i.e. the DB2 interface **DSNELI** (for batch) must also be linked (a source of confusion could occur with CICS/IMS).

   If DL/I and DB2 data bases are concurrently defined in a **QPAC-Batch** program an additional linkage editor step is necessary.
   The module QPACBDB2 that has been link edited in advance has to be additionally link edited under the name QPACBDBI whereby the interface DSNELI has to be replaced by **DFSLI000**.
   The following replacement is included in the linkage editor input stream file 1.

```
//LKED.SYSIN  DD *
 REPLACE DSNAA
 REPLACE DSNELI
 INCLUDE SYSLMOD(QPACBDB2)
 INCLUDE SYSLIB(DFSLI000)
 NAME QPACBDBI(R)
/*
```

Fig. 35: DB2 installation with DL/I under z/OS

3. If DB2 data bases are used without the **TSO batch program IKJEFT01** (Parm definition `PARM=PLAN=...,DB2ID=...`) an additional linkage editor step is necessary.
   The module QPACBDB2 that has been link edited in advance has to be additionally link edited under the name QPACBDBT whereby the interface DSNELI has to be replaced by **DSNALI**.
   The following replacement is included in the linkage editor input stream file 1.

```
//LKED.SYSLIN DD *
 REPLACE DSNAA
 REPLACE DSNELI
 INCLUDE SYSLMOD(QPACBDB2)
 INCLUDE SYSLIB(DSNALI)
 NAME QPACBDBT(R)
/*
```

Fig. 36: DB2 installation without IKJEFT01 under z/OS

4. The DBRM member with a logical record length of 80 (LRECL=80), is the **3rd file** on the QPAC product tape. This member must be stored in the DB2 PDS for DBRM members under the name **QPACBDB2 (QPAC-Batch)** and/or **QPACODB2 (QPAC-Online)**.
The DBRM module contains the name QPAC as CREATOR.

```
// EXEC PGM=QPAC
//STEPLIB DD DISP=SHR,DSN=QPAC.LOADLIB
//QPACLIST DD SYSOUT=*
//IPF DD DSN=QPAC.BATCH.DBRM,DISP=OLD,LABEL=(3,SL),
// DCB=(RECFM=FB,LRECL=80,BLKSIZE=3120),
// VOL=SER=QPBvrm,UNIT=TAPE
//OPF DD DSN=DBRMLIB(QPACBDB2),DISP=OLD (Batch)
or
//IPF DD DSN=QPAC.CICS.DBRM,DISP=OLD,LABEL=(3,SL),
// DCB=(RECFM=FB,LRECL=80,BLKSIZE=3120),
// VOL=SER=QPOvrm,UNIT=TAPE
//OPF DD DSN=DBRMLIB(QPACODB2),DISP=OLD (Online)
//QPACIN DD *
IPF=SQ,WP=WPOS7001
OPF=SQ,WP=WPOS7001
END
/*
```

Fig. 37: Storing the DBRM member QPACBDB2/QPACODB2

5. A BIND must be carried out for the plans that are to be used, in which at least the QPACBDB2 (QPAC-Batch) and/or QPACODB2 (QPAC-Online) programs are specified.

```
//         EXEC PGM=IKJEFT01
//STEPLIB  DD   DSN=DB2.RESLIB, ...
//DBRMLIB  DD   DSN=DBRMLIB,DISP=SHR
//SYSTSPRT DD   SYSOUT=*
//SYSTSIN  DD   *
   DSN   SYSTEM(DB2)
   BIND  PLAN(qpacplan)      -
         MEMBER(QPACBDB2)    -   (Batch)
         or
         MEMBER(QPACODB2)    -   (Online)
         ACT(ADD)            -
         VALIDATE(BIND)      -
         ISOLATION(CS)       -
         ACQUIRE(USE)        -
         RELEASE(COMMIT)
/*
```

Fig. 38: BIND execution with QPACBDB2/QPACODB2

6. A sample RDO CICS DB2TRAN definition:

```
CEDA DEFine DB2TRAN(QPAC)

 DB2TRAN        : QP1C
 Group          : QPAC
 DEscription    :
 Entry          : QP1C
 Transid        : QP1C
```

Fig. 39: Sample RDO CICS DB2TRAN definition

7. A sample RDO CICS DB2ENTRY definition:

```
CEDA DEFine DB2Entry(QPAC)

 DB2Entry       : QPAC
 Group          : QPAC
 DEscription    :
THREAD SELECTION ATTRIBUTES
 TRansid        :
THREAD OPERATION ATTRIBUTES
 ACcountrec     : Txid
 AUTHId         : xxxx
 AUTHType       :
 DRollback      : Yes
 PLAN           : QPACPLAN
 PLANExitname   :
 PRIority       : High
 PROtectnum     :
 THREADLimit    :
 THREADWait     : Pool
```

Fig. 40: Sample RDO CICS DB2ENTRY definition

Chapter 5. DL/I Support Feature

Installation Procedure

1. In order to use the DL/I support feature, the following conditions must be met:

   - DL/I FULL must be installed
   - the DL/I batch processing routines must be stored in the prescribed libraries.

2. The catalog run of the QPAC tape, (which must include the DL/I support feature), must be carried out according to the official LINKAGE EDITOR conventions, following which, the DL/I support feature is available for use.

3. Attention:
   During the **batch** linkage editor run the interface DFSLI000 must be linked for the connection to ASMTDLI.
   During the **CICS online** linkage editor run the interface DFHDLIAI must be linked.
   Please have the correct SYSLIB concatenation in mind.

Chapter 6. Batch-Online Communication via EXCI

The QPAC EXCICS( ) Function

The QPAC-Batch function `EXCICS( )` provides for the communication support between a QPAC-Batch program and QPAC-CICS-Online. Please consult the QPAC-Batch Reference Manual.

Chapter 7. Web Support

Installation Information

RDO Definitions

DOCTEMPLATE: (example)

```
OBJECT CHARACTERISTICS                    CICS RELEASE = 0411
 CEDA  View Doctemplate( EXTRCTF1 )
  Doctemplate    : EXTRCTF1
  Group          : QPAC
  Description    :
 FULL TEMPLATE NAME
  TEmplatename   : EXTRCTF1
 ASSOCIATED CICS RESOURCE
  File           :
  TSqueue        :
  TDqueue        :
  Program        :
  Exitpgm        :
 TEMPLATE SUBLIBRARY
  Library        :
  Membername     : EXTRCTF1
 TEMPLATE PROPERTIES
  Appendcrlf     : Yes                Yes ! No
  TYpe           : Ebcdic             Binary ! Ebcdic
```

Fig. 41: DOCTEMPLATE example

TCPIPSERVICE:

```
OBJECT CHARACTERISTICS                     CICS RELEASE = 0411
 CEDA  View TCpipservice( QPACNSSL )
  TCpipservice   : QPACNSSL
  Group          : QPAC
  Description    :
  Urm            : DFHWBADX
  Portnumber     : 01080              1-65535
  Certificate    :
  STatus         : Open               Open ! Closed
  SSl            : No                 Yes ! No ! Clientauth
  Attachsec      : Verify             Local ! Verify
  TRansaction    : CWXN
  Backlog        : 00005              0-32767
  TSqprefix      :
  Ipaddress      :
  SOcketclose    : No                 No ! 0-240000
```

Fig. 42: TCPIPSERVICE example

CICS Table Definitions

DFHSIT for CWS

```
TCPIP=YES,
WEBDELAY=(5,60)
```

Fig. 43: DFHSIT for CWS

IBM Redbooks

As a very good aid during the implementation of the WEB support we recommend the follwoing IBM redbook:

Architecting Web Support to CICS	SG24-5466

Execution from a Web Browser with URL

```
http://ip.address:portno/CICS/CWBA/QPACONLA?progname
```

ip.address is the unique internet name or IP address of the operating system.
portno is the CWS listener port number defined in RDO TCPIPSERVICE.
CICS is defined if a converter is not required.
CWBA is the supported alias.
QPACONLA is the QPAC entry program to the WEB application.
progname is the application program in the default QPAC sublibrary B.WEBC.*progname*.
Any other sublibrary may be used if you define the full qualification:
QPACONLA?B.*sublib.progname*
e.g. .../QPACONLA?B.WEBS.ANYPROGRAM