

Installing TPF in the Host Environment

The TPF component uses the Adabas SVC to communicate and pass data between the host and target systems. This component and the appropriate Adabas linkage routine must therefore be installed prior to the installation of TPF. Please consult the appropriate Adabas documentation.

This document covers the following topics:

- TPF Facilities Overview
 - Installation under CICS
 - Installation under TSO (MVS only)
 - Installation under Com-plete
 - What Next?
-

TPF Facilities Overview

TPF provides a number of facilities to allow you to customize the product to your site's requirements. These facilities include:

- The ACSTAB table;
- The Direct Call feature;
- Printing from TPF;
- The Transparency Feature;
- Reverse Access programming.

This section provides a brief outline of these facilities.

- **The ACSTAB table:**
Site-specific items on the host side of an TPF installation can be specified in the ACSTAB table. These include, for example, the Adabas SVC number, DBID and the linkage routine to be used.
- **The Direct Call feature:**
This feature enables the user to define transactions that are automatically routed to the TPF target system, where a specific program or transaction is started. The Direct Call feature is described in detail in the section **TPFXTAB Subtable** in **The ACSTAB Table**.
- **Printing from TPF:**
CICS sites can define a local printer in the TPF target system's TIBTAB. Printouts from the target system are printed on the corresponding CICS local printer.
- **The Transparency feature:**
Transparency is a feature of TPF which simplifies the conversation between a Natural CICS/TSO environment and an environment in which Natural runs under TPF. The transparency feature

allows Natural programs which formerly ran under CICS or TSO to run under TPF with no changes. More details on transparency are given in *TPF Transparency under CICS and TSO*.

- **Reverse Access:**

Many Natural programs currently under CICS call 3GL subroutines to perform certain functions. If these subroutines do not directly use any CICS facilities, then they can simply be installed to run in the TPF address space/partition. However, if a 3GL subroutine uses CICS for any reason, (for example, EXEC CICS statements), then it must run in the CICS address space/partition. Reverse Access provides this capability as a feature of TPF transparency, and is described in *TPF Transparency under CICS and TSO*.

Installation under CICS

To install the TPF component under CICS, proceed as follows:

Step 1: Unload the Installation Data Set

- Unload the COMvrs.SCRT and COMvrs.LOA3 data sets from the installation tape. Please consult the report from the tape creation system to determine the order of the files on the installation tape. The COMvrs.LOA3 library contains the TPF load modules required for CICS.

On VSE systems, note that the required library is the TPFvrs sublibrary unloaded in the installation procedure using the sample JCS JCLINST1.

Step 2: Modify the ACSTAB Table

- **The ACSTAB Table** contains a full description of the ACSTAB table, An example ACSTAB is provided in the source library called ACSTAB, use this and modify it to suit your installation requirements. The table is subdivided in four parts. Each part can be modified to suit your own installation requirements.
 1. ACSDEF describes the ACCESS Default Table which is used to define data used by the host node. This subtable must be unique for each host region.
 2. ACSTBL describes the ACCESS Node Table which lists the various target systems available for communication with the host system and also the host system with an entry for every CICS host node.
 3. ACSSCHC describes the ACCESS screen-to-hardcopy table which defines the hardcopy device printer where printout will be routed for every terminal in the system.

Note:
If you do not require the screen-to-hardcopy function, then you need not specify this subtable.
 4. TPFXTAB describes the Transparency table which describes the various transactions you can use to enter your TPF environment and the way you give user ID, password and data to your TPF.

Step 3: Assemble and Link the ACSTAB Table

- After modification, assemble and link the ACSTAB table using the sample job from the source library (JCLASMTB).

Step 4: Link the CICS Modules

- Using the sample link job in the source library (JLLNKCI), link the TPF modules to a load library which is concatenated in the CICS start-up procedure. You are recommended *not* to use the load library you loaded from the tape.

Step 5: Define the CICS Environment

1. Include the TPF modules in the program tables or define them with the CICS CEDA online transaction.

The following entries must be defined in the PPT or with CEDA.

```
DFHPPT TYPE=ENTRY , PROGRAM=ACCESS46 , RES=YES
DFHPPT TYPE=ENTRY , PROGRAM=ACSBEG46
DFHPPT TYPE=ENTRY , PROGRAM=ACSSPL46
DFHPPT TYPE=ENTRY , PROGRAM=ACSTAB46
DFHPPT TYPE=ENTRY , PROGRAM=TPFXF46
DFHPPT TYPE=ENTRY , PROGRAM=ACSUCTR )see note 4
```

The following entries must be defined in the PCT or with CEDA.

```
DFHPCT TYPE=ENTRY , PROGRAM=ACCESS46 , *
      TRANSID=XT46 , TWASIZE=250 )see note 1
DFHPCT TYPE=ENTRY , PROGRAM=TPFXF46 , *
      TRANSID=ADMN , TWASIZE=250 )see note 2
DFHPCT TYPE=ENTRY , PROGRAM=TPFXF46 , *
      TRANSID=ATPF , TWASIZE=250 )see note 2
DFHPCT TYPE=ENTRY , PROGRAM=TPFXF46 , *
      TRANSID=BTPF , TWASIZE=250 )see note 2
DFHPCT TYPE=ENTRY , PROGRAM=ACSBEG46 , *
      TRANSID=AB46 , TWASIZE=250 )see note 3
DFHPCT TYPE=ENTRY , PROGRAM=ACSSPL46 , *
      TRANSID=AP46 , TWASIZE=360 )see note 3
DFHPCT TYPE=ENTRY , PROGRAM=ACSUCTR ,
      TRANSID=ACSU , TWASIZE=250 )see note 4
```

Notes:

1. 'XT46' or the TRANSID transaction ID specified in the ACSDEF sub-table.
2. Plus any other user transactions defined in the TPFXTAB sub-table.
3. 'AB46' and 'AP46' are reserved transaction IDs for printing under TPF.
4. These entries are only required if the upper case translation feature is to be implemented. The program ACSUCTR and transaction ACSU are redundant under CICS 3.3 and above, and should not be installed (see the section on upper case translation in the section **The ACSTAB Table** in this chapter).

If the transaction AB46 and AP46 are not defined, then the terminal functions of TPF will be available as normal but printing functions will not be available.

2. In order to automatically activate printer support each time CICS starts, include program ACSBEG46 in the PLT:

```
DFHPLT TYPE=ENTRY,PROGRAM=ACSBEG46
```

- If ACSBEG46 is not defined in the PLT, printing can be started by starting transaction AB46 from the terminal.

More information about TPF printing is given in the section **Printing under TPF**.

- If CEDA was not used to define the environment, the PLT, PPT and PCT tables must now be assembled and linked in the usual manner, and the CICS system must be restarted.
- If CEDA was used, then the group defined has to be installed.

Step 6: Verify the Installation and Log On

- The installation of the TPF component is now finished.

To test the installation, use the ADMN transaction, which logs on to TPF using the predefined user ID SAGADMIN and password ADMIN:

```
ADMN ( , ADMIN)
```

This user ID has a definition on the system data set with the required authorization to define more user IDs to the system using the user ID maintenance facilities of UUTIL (see the Utilities documentation). The initial password required by UUTIL is PASSWORD, unless otherwise specified by the sysparm ULOGM.

After defining other user IDs (see the section **User ID Considerations**), you are recommended to change the password for SAGADMIN and use the user ID SAGADMIN in emergency cases only.

Installation under TSO (MVS only)

To install the TPF component under TSO in MVS environments, proceed as follows:

Step 1: Unload the Installation Data Set

- Unload the COMvrs.SRCT and COMvrs.LOA3 libraries from the installation tape.

Please consult the report from the tape creation system in order to determine the order of the files on the tape. The load library contains the relevant load modules for TSO.

Step 2: Modify the ACSTAB Table

- For a full description of the ACSTAB table see the section **The ACSTAB Table**. An example ACSTAB is specified in the source library called ACSTAB, use this and modify it to suit your installation.

The table is subdivided in four parts.

1. ACSDEF describes the ACCESS Default Table which is used to define data used by the host node. This subtable must be unique for each host region.
2. ACSTBL describes the ACCESS Node Table which lists the various target systems available for communication with the host system and also the host system with an entry for every TSO host node.
3. ACSSCHC describes the ACCESS screen-to-hardcopy table. This is not required for TSO.
4. TPFXTAB describes the Transparency table which describes the various transactions you can use to enter your environment and the way you give user ID, password and data to your system.

Each part can be modified to suit your own installation requirements.

Step 3: Assemble and Link the ACSTAB Table

- After modification, the ACSTAB table must be assembled and linked using the sample job from the source library (JCLASMTB).

Step 4: Link the TSO Modules

- Using sample job JCLLNKTS, link the TPF module to an appropriate TSO library. Do *not* use the load you loaded from tape.

Step 5: Create the TSO Environment

- Create and save a CLIST in an appropriate TSO library using the sample given in member JCLCLIST. This member includes two useful examples of a simple way to invoke TPF under TSO.

The TSORUN member supplied consists of a single statement which is used to control the TSO TPF programs. Using this simple ADARUN card the SVC number of the Adabas SVC and the Adabas runtime options can be specified. Please modify the Adabas SVC number in the supplied TSORUN member to suit your installation.

Note that TPF under TSO communicates using the Adabas SVC via the link routine ADALNK. The module TPFXTSO is linked with the Adabas module ADAUSER, the first time ADAUSER is called it loads module ADARUN which subsequently loads the link routine ADALNK to issue the SVC request.

The ADARUN card, the ADALNK, ADARUN and ADAUSER modules must therefore be accessible at run-time.

TPFXTSO will load the ACSTAB during initialization so that ACSTAB must also be accessible.

Step 6: Verify the Installation and Log On

- The installation on the host side is now finished. To test the installation, execute the CLIST created.

To log on directly to TPF with the predefined userid SAGADMIN, use the optional keywords `trandid`, `nodeid`, `userid` and `password` in the CLIST.

```
EX 'xx.xx.xx(clist)' 'TRANID(ADMN) NODEID (COM51) USERID(SAGADMIN)
PASSWORD(ADMIN)'
```

See the example job in the source library (JCLCLIST).

This user ID has a definition on the system data set with the required authorization to define more user IDs to the system using the user ID maintenance facilities of UUTIL (see the Utilities documentation). The initial password required by UUTIL is PASSWORD, unless otherwise specified by the sysparm ULOGM.

After defining other user IDs (see *User ID Considerations*), you are recommended to change the password for SAGADMIN and use the user ID SAGADMIN in emergency cases only.

Installation under Com-plete

To install the TPF component under Com-plete, proceed as follows:

Step 1: Unload the Installation Data Set

- Unload the COMvrs.SRCT and COMvrs.LOA3 libraries from the installation tape.

Please consult the report from the tape creation system in order to determine the order of the file on the tape. The load library contains the load modules for Com-plete.

On VSE systems, note that the host.load library is the TPFvrssublibrary unloaded in the installation procedure using the sample JCS JCLINST1.

Step 2: Modify the ACSTAB Table

- For a full description of the ACSTAB table see the section **The ACSTAB Table**. An example ACSTAB is specified in the source library called ACSTAB, use this and modify it to suit your installation.

The table is subdivided in four parts.

1. ACSDEF describes the ACCESS Default Table which is used to define data used by the host node. This subtable must be unique for each host region.
2. ACSTBL describes the ACCESS Node Table which lists the various target systems available for communication with the host system and also the host system with an entry for every TSO host node.
3. ACSSCHC describes the ACCESS screen-to-hardcopy table. This is not required for Com-plete.
4. TPFXTAB describes the Transparency table which describes the various transactions you can use to enter your environment and the way you give user ID, password and data to your system.

Each part can be modified to suit your own installation requirements.

Step 3: Assemble and link the ACSTAB Table

- After modification the ACSTAB table must be assembled and linked using the example job from the source library (JCLASMTB).

Step 4: Link the Module for Complete

- With the example job JCLLNKCO, link the TPF module to an appropriate load library specified in the COMPLETE start-up procedure. Please do not link the module to the load you loaded from tape.

Step 5: Create the Complete environment

- Catalog the linked TPF module with ULIB using the following command:

```
ULIB CAT,TPF46,RG=32K,PV
```

Step 6: Verify the Installation and Log On

- The installation on the host side is now finished. To test the installation, call the module TPF46 from the USTACK menu. Log on directly to TPF with the predefined user ID SAGADMIN and password ADMIN.

This user ID has a definition on the system data set with the required authorization to define more user IDs to the system using the user ID maintenance facilities of UUTIL (see the Utilities documentation). The initial password required by UUTIL is PASSWORD, unless otherwise specified by the sysparm ULOGM.

After defining other user IDs (see the section **User ID Considerations**), you are recommended to change the password for SAGADMIN and use the user ID SAGADMIN in emergency cases only.

What Next?

TPF is now up and ready for work. How you continue depends on whether you have installed TPF for the first time or whether you are migrating from a previous version.

If you have installed TPF for the first time, no further migration is necessary. You can continue with customization steps described in the System Programming documentation and the Utilities documentation.

If you are going to use the model user ID, SYSCOM, then you will need to modify the definition of this user to suit your requirements (via the UUTIL subfunction UM).