Transaction Processing Facility

General Information

Version 4 Release 1

GH31-0147-00

Transaction Processing Facility

General Information

Version 4 Release 1

GH31-0147-00

Note!
Before using this information and the product it supports, be sure to read the general information under “Notices” on page vii.

First Edition (September 1993)

This edition applies to Version 4 Release 1 Modification Level 0 of IBM Transaction Processing Facility, program number 5748-T14, and to all subsequent releases and modifications until otherwise indicated in new editions or technical newsletters. Make sure you are using the correct edition for the level of the product.

IBM welcomes your comments. Address your comments to:

IBM Corporation
TPF Systems Information Development
Mail Station P923
2455 South Road
Poughkeepsie, NY 12601-5400
USA

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

Contents

Figures

Notices

References in this book to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service in this book is not intended to state or imply that only IBM's product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any of IBM's intellectual property rights may be used instead of the IBM product, program, or service. Evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM, is the user's responsibility.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
USA

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
Department 830A
Mail Drop P131
2455 South Road
Poughkeepsie, NY 12601-5400
USA

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

Any pointers in this book to non-IBM Web sites are provided for convenience only and do not in any way serve as an endorsement. IBM accepts no responsibility for the content or use of non-IBM Web sites specifically mentioned in this book or accessed through an IBM Web site that is mentioned in this book.

Trademarks

The following terms are trademarks of the IBM Corporation in the United States or other countries or both:

ACF/VTAM
CICS
DATABASE 2
DB2
Distributed Relational Database Architecture
DRDA
Enterprise Systems Connection Architecture
EOCF/2
ES/3090
ES/9000
ESA/370

ESA/390
ESCON
IBM
MVS/ESA
Operating System/2
OS/2
SAA
Sysplex Timer
Systems Application Architecture
VTAM

Other company, product, and service names may be trademarks or service marks of others.

About This Book

This book provides a basic understanding of the IBM Transaction Processing Facility (TPF) Version 4 Release 1 system as well as an overview of the TPF operating system, describing its characteristics, functions, and the features that allow the TPF system to grow. This book helps your installation answer the question: “Should we buy the TPF 4.1 system?”

In this book, abbreviations are often used instead of spelled-out terms. Every term is spelled out at first mention followed by the all-caps abbreviation enclosed in parentheses; for example, Systems Network Architecture (SNA). Abbreviations are defined again at various intervals throughout the book. In addition, the majority of abbreviations and their definitions are listed in the master glossary in the *TPF Library Guide*.

Who Should Read This Book

This book is intended for executives, managers, and system analysts to help them decide if the TPF system is suitable for their environment. It is also suitable for anyone who needs a general understanding of the TPF system.

The reader should have a basic knowledge of data processing and data communications.

How This Book is Organized

This book provides background information describing the TPF system, its concepts, structure, and features, and explains in detail the new functions in the TPF 4.1 system.

Conventions Used in the TPF Library

The TPF library uses the following conventions:

Conventions	Examples of Usage
italic	Used for important words and phrases. For example: A *database* is a collection of data. Used to represent variable information. For example: Enter **ZFRST STATUS MODULE** *mod*, where *mod* is the module for which you want status.
bold	Used to represent text that you type. For example: Enter **ZNALS HELP** to obtain help information for the ZNALS command. Used to represent variable information in C language. For example: **level**

Conventions	Examples of Usage

monospaced	Used for messages and information that displays on a screen. For example:
	`PROCESSING COMPLETED`
	Used for C language functions. For example:
	`maskc`
	Used for examples. For example:
	`maskc(MASKC_ENABLE, MASKC_IO);`

bold italic	Used for emphasis. For example: You ***must*** type this command exactly as shown.
Bold underscore	Used to indicate the default in a list of options. For example: **Keyword=OPTION1** | **DEFAULT**
Vertical bar |	Used to separate options in a list. (Also referred to as the OR symbol.) For example: **Keyword=Option1** | **Option2** **Note:** Sometimes the vertical bar is used as a *pipe* (which allows you to pass the output of one process as input to another process). The library information will clearly explain whenever the vertical bar is used for this reason.
CAPital LETters	Used to indicate valid abbreviations for keywords. For example: KEYWord=*option*

Scale	Used to indicate the column location of input. The scale begins at column position 1. The plus sign (+) represents increments of 5 and the numerals represent increments of 10 on the scale. The first plus sign (+) represents column position 5; numeral 1 shows column position 10; numeral 2 shows column position 20 and so on. The following example shows the required text and column position for the image clear card.
	`|...+....1....+....2....+....3....+....4....+....5....+....6....+....7...`
	`LOADER   IMAGE CLEAR`
	Notes:
	1. The word LOADER must begin in column 1.
	2. The word IMAGE must begin in column 10.
	3. The word CLEAR must begin in column 16.

Related Books

A list of related books for the TPF 4.1 announcement follows:

- *TPF Migration Guide: Program Update Tapes* (to be available prior to the general availability of the TPF 4.1 system.)
- *TPF Concepts and Structures*.

The following is an IBM Systems Application Architecture (SAA) book that can be purchased separately:

- *SAA Common Programming Interface/C Reference*.

How to Send Your Comments

Your feedback is important in helping to provide the most accurate and highest quality information. If you have any comments about this book or any other TPF information, use one of the methods that follow. Make sure you include the title and number of the book, the version of your product and, if applicable, the specific location of the text you are commenting on (for example, a page number or table number).

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

- If you prefer to send your comments electronically, do either of the following:
  - Go to http://www.ibm.com/tpf/pubs/tpfpubs.htm.

    There you will find a link to a feedback page where you can enter and submit comments.
  - Send your comments by e-mail to tpfid@us.ibm.com
- If you prefer to send your comments by mail, address your comments to:

  IBM Corporation
  TPF Systems Information Development
  Mail Station P923
  2455 South Road
  Poughkeepsie, NY 12601-5400
  USA
- If you prefer to send your comments by FAX, use this number:
  - United States and Canada: 1 + 845 + 432 + 9788
  - Other countries: (international code) + 845 + 432 +9788

Overview of the TPF System

This chapter introduces the Transaction Processing Facility (TPF) operating system, describing its characteristics, functions, and the features that permit the TPF system to grow.

What Is the TPF System?

The Transaction Processing Facility (TPF) system is an operating system that works with application programs to process transactions for customers in a real-time environment. For example, the TPF system can be used for computing needs such as transaction processing, network switching, and front-end processing.

The TPF system is designed for businesses and organizations that have high online transaction volumes and large networks. Because of its high capacity and high availability characteristics, the TPF system is well-suited for environments where growth is expected to be very fast or unpredictable, or where there are high peak periods of transaction activity. It is especially useful for applications that need high capacity and extremely low cost per transaction. Figure 1 shows the types of user requests and data requirements that are handled by the TPF system.

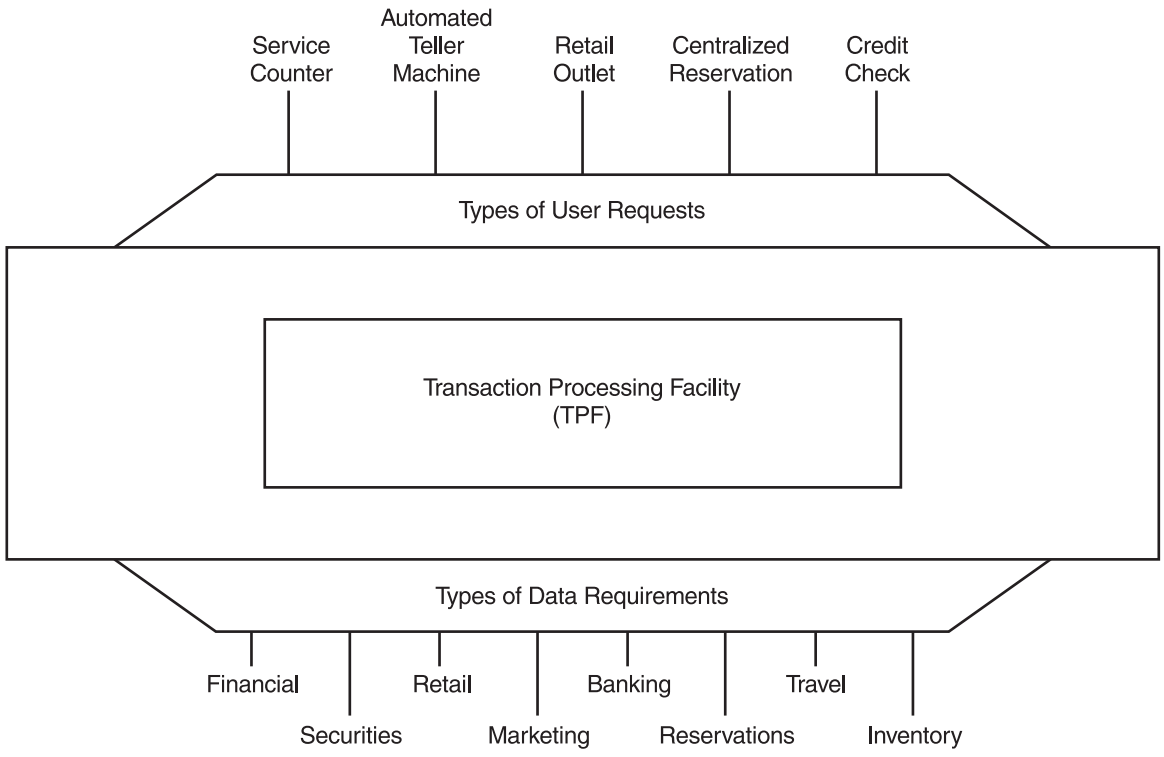

Figure 1. Applications Using the TPF System

TPF System Characteristics

The TPF system can help your organization give customers prompt and accurate service by providing:

- Message-driven processing
- Very high throughput with quick response time
- High availability

- Data integrity
- Low cost per transaction.

Message-Driven Processing

The TPF system is a message-driven system; each query that the TPF system receives is called a message. A transaction refers to the series of message pairs (input/reply) necessary for completing a customer's request. A business transaction can consist of multiple message pairs such as a cash withdrawal from an automated teller machine (ATM), or an airline reservation with advance seating. The number of message pairs required to complete a transaction depends on the application design and the complexity of the specific information required by the agent. For example, a credit verification often results in a transaction of just one message pair.

Very High Throughput with Quick Response Time

The TPF system provides very high throughput measured by the number of messages per second. A single transaction usually involves several messages and multiple database accesses.

The typical TPF system processes several hundred messages per second. For even higher throughput, a loosely coupled TPF system can process thousands of messages per second.

You can obtain more than 5000 messages per second depending on your configuration. The maximum throughput in a fully configured loosely coupled complex is unknown because no tool exists that is capable of testing the TPF system's maximum performance. Your system's actual capacity with the TPF system depends on the hardware configuration and the application design.

The response time of the TPF system within a network is typically less than three seconds from the time the user sends a message to the time the user receives a response to the message. During that 3 seconds, the message typically spends approximately 0.5 seconds in the TPF system.

High Availability

The TPF system provides very high system availability. In many cases, the scheduled availability has exceeded 99.9%, which represents fewer than 10 minutes of downtime per week. Scheduled availability is measured in terms of a system processing work 24 hours per day, 7 days per week, minus the time for scheduled outages. The TPF system can restart in one-half minute to 2 minutes of a system failure, so the end user is often not even aware of the failure.

Even higher availability can be achieved when TPF central processing complexes (CPCs) are run in a loosely coupled configuration. The system can be designed so that the failure of any one CPC will be undetected by most end users. Additional CPCs can be brought into the online complex without any outages.

Having a properly configured backup system or backup processor can increase the scheduled availability to greater than 99.9%. TPF customers who have single central processing complex (CPC) systems usually have another CPC that is capable of taking over from the primary one. Usually, this backup system is assigned to run jobs that can be interrupted if necessary. Typically, an installation can switch over to the backup CPC within minutes of the primary CPC failure.

Data Integrity

The introduction of virtual address spaces in the TPF environment has significantly increased the integrity of the data environment in the TPF system.

The TPF system also provides the basic tools your installation might need for additional data integrity and recovery. The TPF system reaches its high transaction rate in part by allowing installations to determine the degree of integrity needed for each system resource and to build their own recovery.

Low Cost Per Transaction

Very short path lengths for critical system services such as DASD I/O help to make efficient use of the TPF system's resources and to provide a very low cost per transaction.

TPF System Application Types

The TPF system can be used by applications in many different ways, for example, as a transaction processing system, a network switch, a front-end processor, or all three types at once. Some application programs that work with the TPF system are:

- Reservations
- Credit checks
- Financial applications
- Inventory
- Electronic mail.

Applications for the TPF system can be written in IBM C/370 language or in IBM High Level Assembler/MVS & VM & VSE language.

Transaction Processing

Figure 2 on page 4 shows the use of the TPF central processing complex (CPC) in a transaction processing application. After a message arrives from the network, it is processed by the TPF system, the database is updated, and a message is sent back to the originator of the incoming message. Transaction processing with the TPF system provides high throughput, quick response time, high availability, and low cost per transaction.

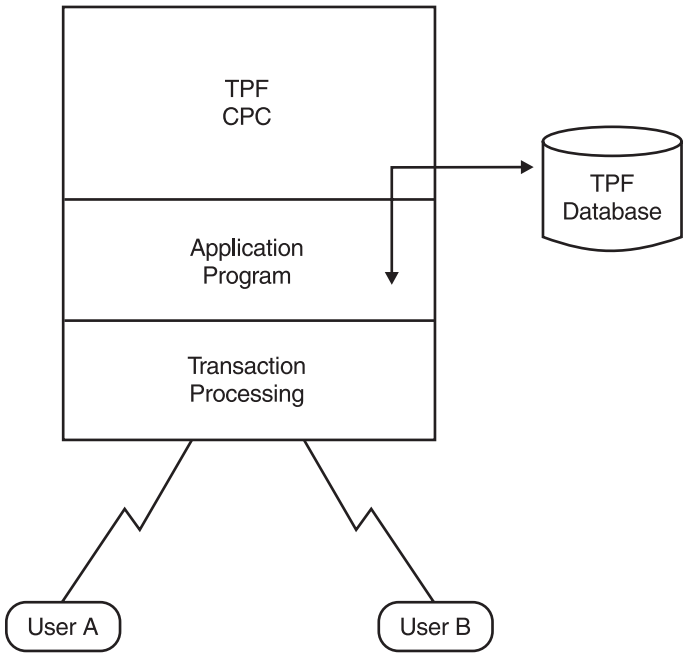

Figure 2. Transaction Processing Application

Network Switching

Figure 3 on page 5 shows how the TPF system is used in a network switching application. This TPF CPC has communication links to other transaction processing systems. An ATM, attached to CPC 1 in Processing Center A, sends a request to CPC 0, the TPF network switch. Using the routing database, TPF switches the request to CPC 2 in Processing Center B. Then, CPC 2 sends a reply through the TPF system, back to the originating ATM. Usually, network switches must distribute messages across many communication lines to many processing centers. Network switching with the TPF system provides rapid routing of messages over communication lines to many different processing centers.

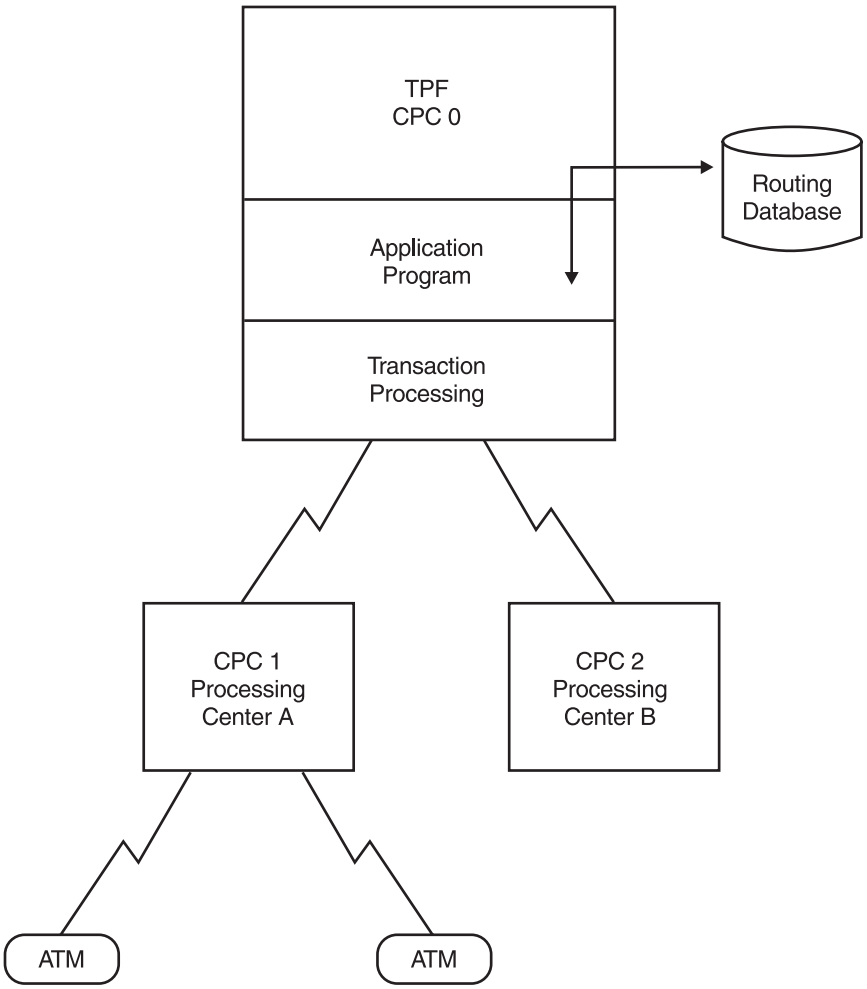

Figure 3. Network Switching Application

Front-End Processing

Figure 4 on page 6 shows a TPF front-end processor application. The advantage of a front-end processor is that it can be programmed to provide temporary services through stand-in applications if the appropriate back-end processor is not available. For example, a credit card check can usually involve looking up the credit limit and existing balances on a database belonging to one of the back-end systems. However, if that system is unavailable, the TPF front-end system can be designed to provide minimal credit checks so business can continue.

Figure 4 on page 6 also shows separate systems with different applications connected to a TPF front-end processor. The front-end processor can direct messages received from the network to the appropriate application processor. The replies from the back-end transaction processors are transmitted back through the TPF system to the source of the message. If the processing center grows, additional back-end processors can be added. Environments that typically use front-end processors include the banking and securities industries as well as some large airline reservation systems.

Front-end processing with the TPF system provides:

- Network connectivity (including concealment of the network from back-end processor)
- A single system image
- Load balancing between back-end processors when the same type of message is being processed
- Temporary processing if back-end processors are not available.

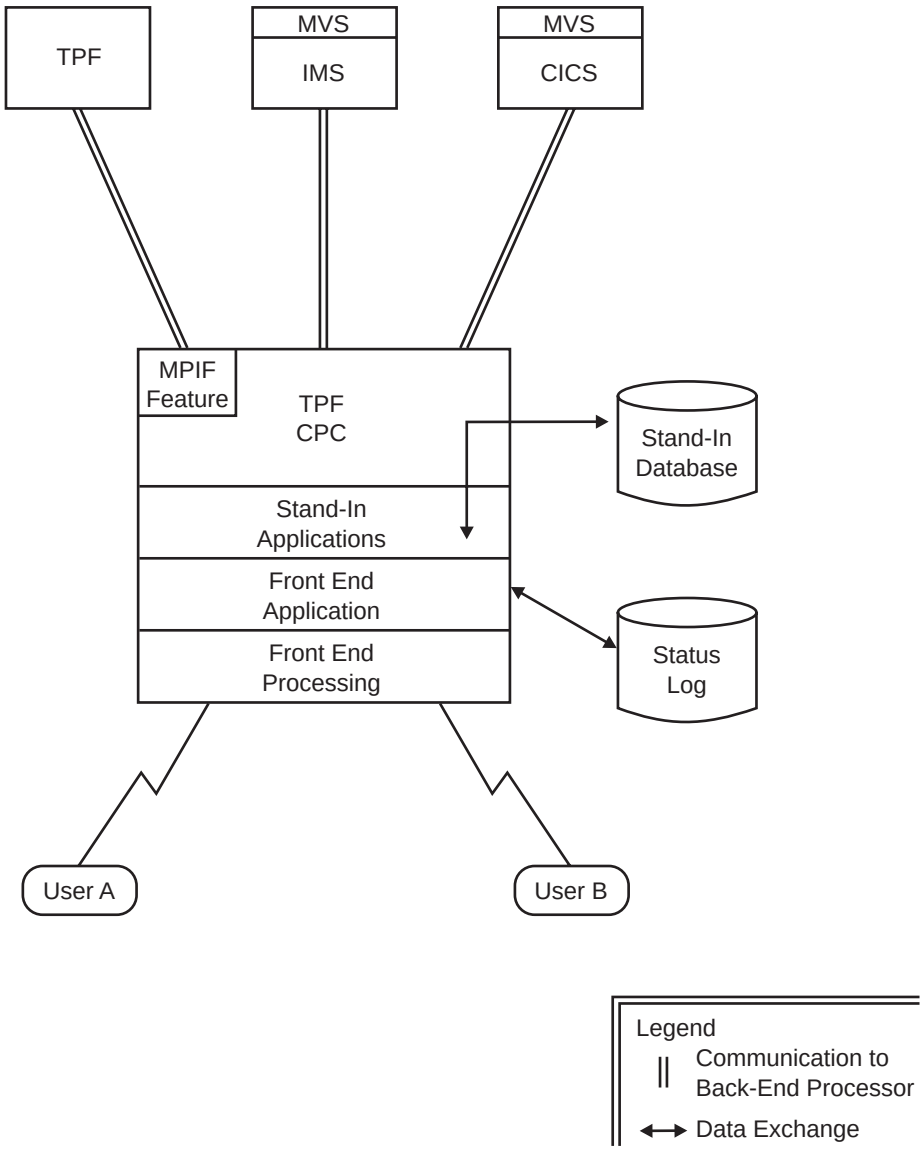

Figure 4. Front-End Processor Application

System and Application Growth

To provide for system and application growth, the TPF system can increase its capacity through different hardware configurations. It can also update one or multiple databases, and update relational databases directly.

The TPF system allows open-ended growth by providing:

- Short path lengths for system services

- The ability to couple up to 8 multiprocessor CPCs together with virtually no increased system overhead
- The ability to communicate between TPF complexes at high speeds using the Multi-Processor Interconnect Facility (MPIF) feature.

The remainder of this chapter describes the TPF functions and features that allow the TPF system to grow in capacity and connectivity.

Configurations

The TPF system can run in a single central processing complex (CPC) or in a multiple CPC environment. A single CPC can have one (uniprocessor) or more (multiprocessor) processors. The TPF system runs on selected models of ES/3090 and ES/9000 processors. TPF configurations can be tightly coupled or loosely coupled.

A *tightly coupled* configuration has a single multiprocessor in which the processors are synchronized to share a single main storage. When the TPF system runs with one image of the TPF system for each CPC, regardless of the number of processors, it is referred to as a tightly coupled configuration.

A *loosely coupled* configuration has two or more CPCs that share a common set of DASD. Sharing a database improves the TPF system's capacity and availability. Through its loosely coupled (LC) facility, the High Performance Option (HPO) feature allows the TPF system to run in a loosely coupled configuration. When using HPO, a maximum of 8 CPCs can share a common database in a loosely coupled configuration.

Figure 5 shows an example of vertical growth from a uniprocessor CPC to a multiprocessor CPC.

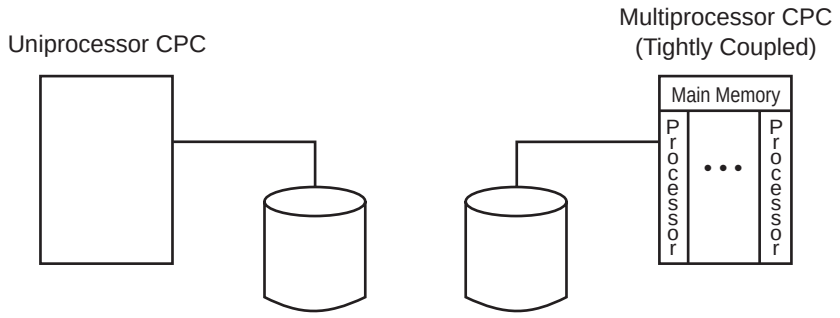

Figure 5. TPF System Capacity for Vertical Growth

Figure 6 on page 8 shows an example of horizontal growth that uses the HPO feature to share the same database between multiple CPCs.

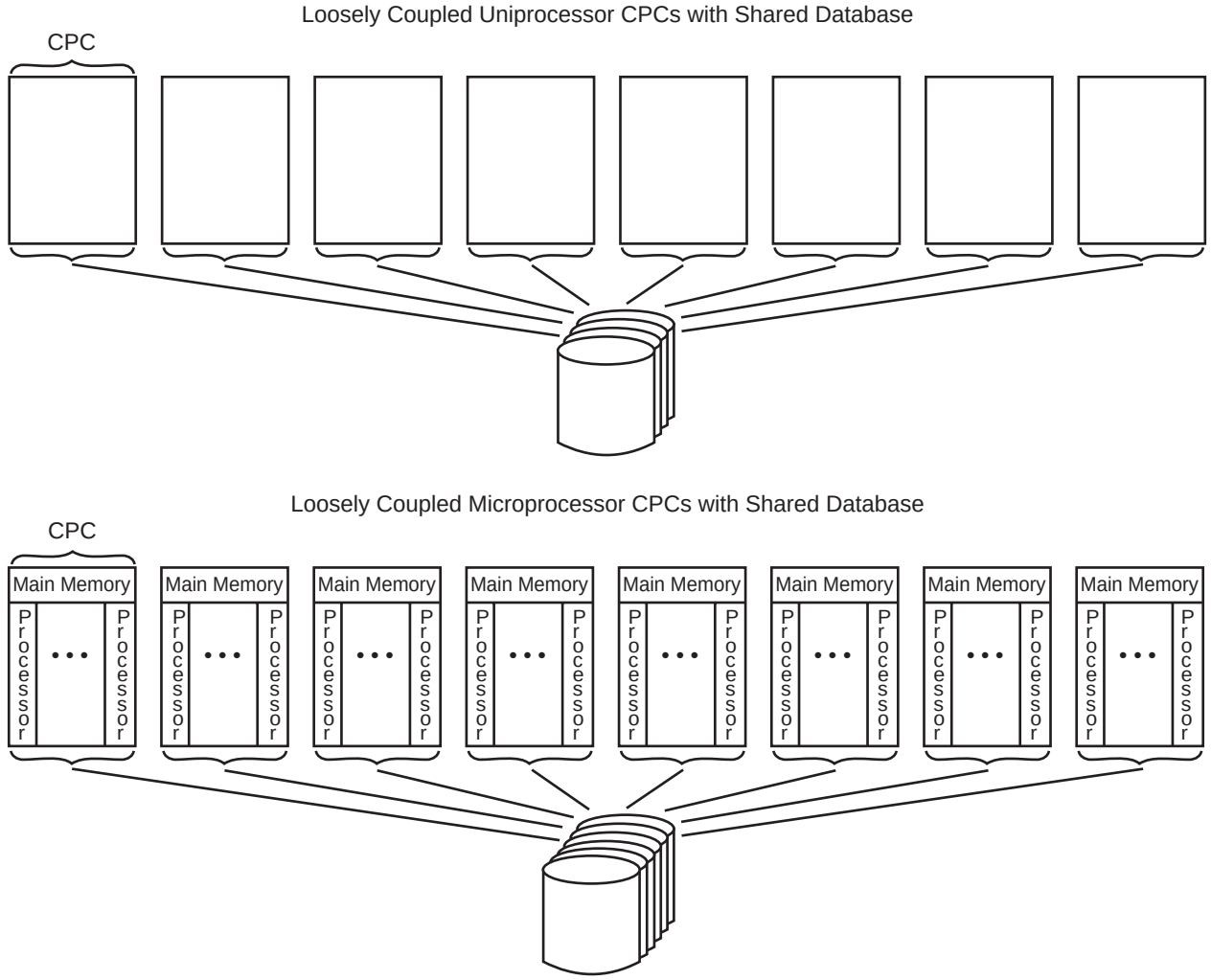

Figure 6. TPF System Capacity for Horizontal Growth

Loosely coupled TPF configurations provide the following advantages over single CPC systems:

- Increased processing capacity

  The capacity of the TPF system increases linearly with each CPC added to a loosely coupled configuration. Coupling one CPC to an existing CPC effectively doubles the capacity; coupling 2 CPCs to an existing CPC effectively triples the capacity. For example, if the capacity of a single ES/9000 CPC is not sufficient, up to eight ES/9000 CPCs can be loosely coupled together. In a loosely coupled configuration, any application can run in any CPC, and any CPC can update any part of the shared database.
- Increased system availability

  Multiple CPCs share the transaction processing workload and maintain the same database, as shown in Figure 7 on page 9.
- More efficient and effective use of computing resources

  Based on workload, the number of CPCs can be dynamically expanded or contracted without affecting system availability. Periodic functions such as system utilities constitute a significant drain on system resources. A CPC can be dedicated to one or several utilities while the remainder of the complex continues transaction processing unaware of concurrent utility activity.

Data integrity in a loosely coupled configuration is ensured through TPF-unique functions for IBM CPCs and disk control units. The disk control units support TPF locking functions that maintain locks on a piece of data so that multiple CPCs cannot update that data simultaneously.

The TPF system uses the hardware timers to synchronize CPC clocks throughout a loosely coupled complex.

The CPCs in a loosely coupled configuration communicate status information to each other at a very high speed using the Multi-Processor Interconnect Facility (MPIF) feature.

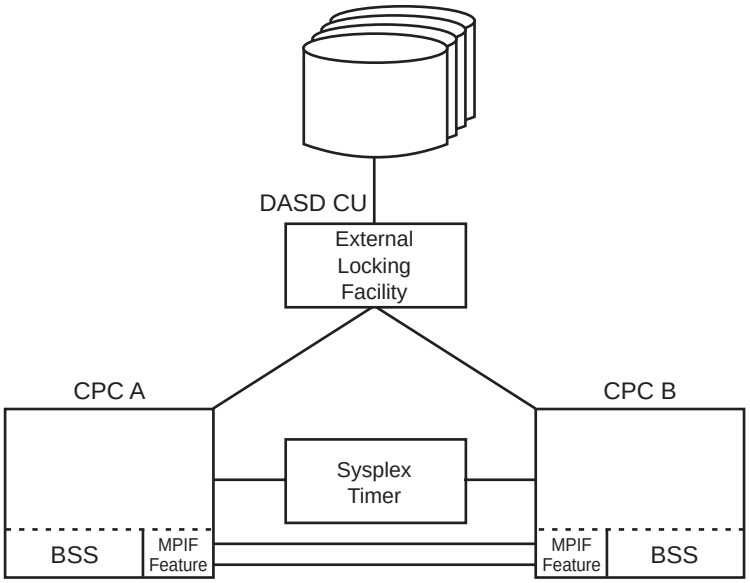

Figure 7. Loosely Coupled Configuration

Connecting to Other TPF System and MVS Configurations

Multiple TPF CPC configurations can be formed through:

- A loosely coupled configuration
- Network control program (NCP) channels.

When CPCs are connected through a loosely coupled configuration, the Multi-Processor Interconnect Facility (MPIF) feature provides an interface that allows TPF configurations to communicate at very high speeds with another TPF configuration through channel-to-channel (CTC) connections. In a loosely coupled configuration, multiple CPCs can share a database.

The MPIF feature allows one TPF CPC to request a service that is received, processed, and returned by another TPF CPC. The requesting CPC need not be aware that a different CPC provided the requested service. The MPIF feature uses CTC connections to communicate in a single loosely coupled CPC or between multiple CPCs. The TPF 4.1 system can communicate with the TPF 3.1 system using the MPIF feature.

Figure 8 on page 10 shows an example of connecting configurations using the MPIF feature and SNA CTC.

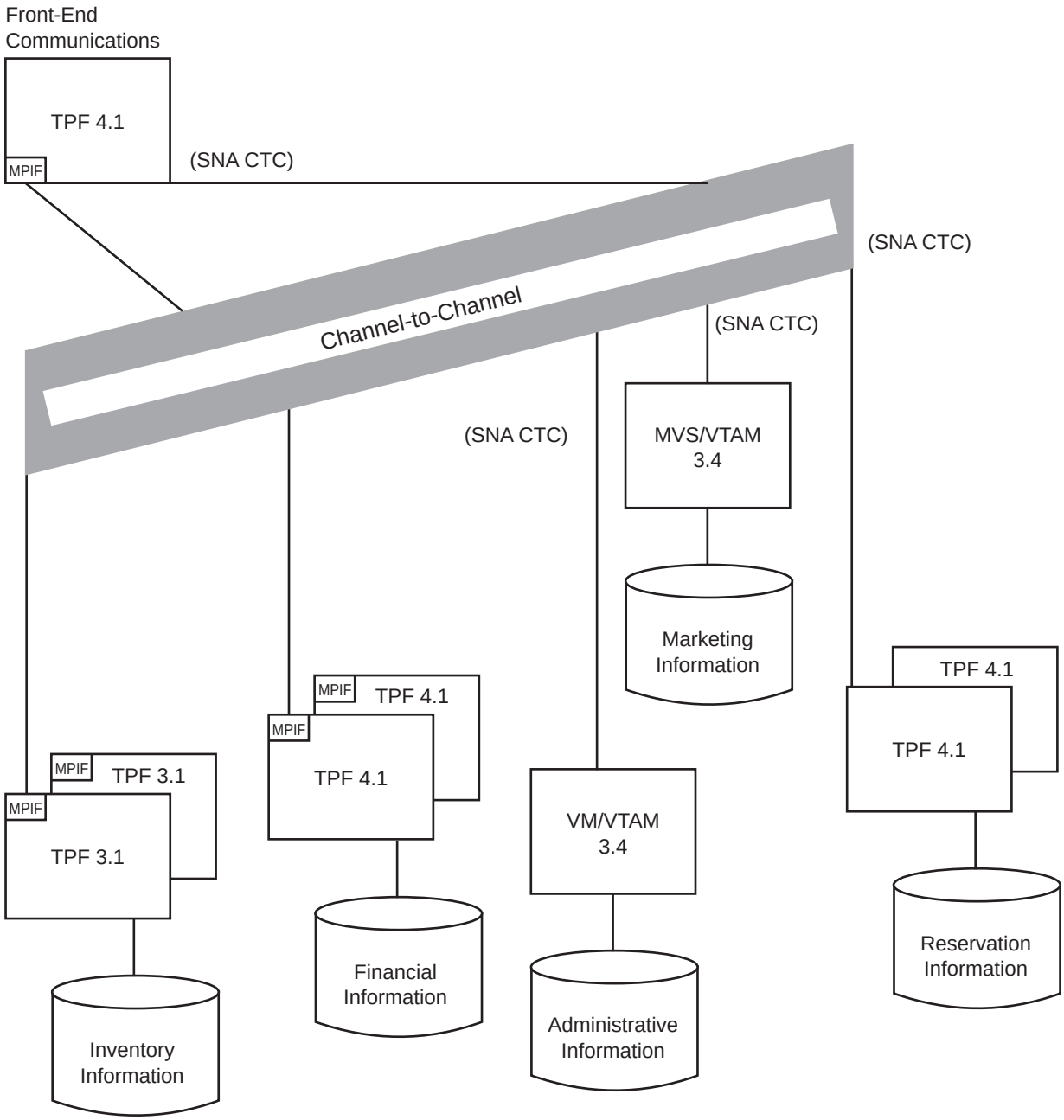

Figure 8. Connecting Configurations Using The MPIF Feature and SNA CTC

Database Support

Through its Multiple Database Function (MDBF), the High Performance Option (HPO) feature allows your installation to run multiple applications concurrently under one TPF system with a unique database for each application.

MDBF extends the TPF system's capabilities by supporting multiple unique databases on the same central processor complex (CPC). Your installation can define up to 64 unique databases or subsystems. A group of applications that share

a database is treated as a unique TPF system. In a TPF system with multiple subsystems, a primary or basic subsystem (BSS) is required for operational control of all TPF system resources.

The BSS contains all system routines, the control program, and supporting programs. The BSS can also include application programs. The other subsystems contain all resources necessary for a stand-alone system except the control program and its supporting programs. Each subsystem relies on the BSS running all system-related tasks, including input and output message processing.

Subsystems are independent and their data is protected from other subsystems. This assures that new applications and equipment can be assimilated in a planned, orderly manner without affecting existing databases.

Application Support

TPF applications can be written in IBM High Level Assembler/MVS & VM & VSE or C language. TPF application programs can directly access relational databases through structured query language (SQL) commands using the TPF Application Requester (TPFAR) feature.

IBM C/370 Language Support

IBM C/370 language is included in the Systems Application Architecture (SAA) Common Programming Interface. IBM C/370 language support and the IBM C/370 Compiler program product provide support for the C/370 language. The use of this high-level, general purpose language can increase productivity and quality in application programming shops. By writing application programs in the C/370 language, your installation can create fast and compact code with all the benefits of coding in a high-level language.

TPF C/370 language support includes the following:

- The IBM C/370 Compiler's TPF system option results in the generation of object code consistent with TPF application requirements.
- Connection to permit C application programs to interact with each other, with C library functions, and with assembler language programs. A "quick enter routine" provides fast connection between C application programs and C library functions.
- A set of TPF application interfaces written specifically for C programs and implemented as C functions.
- A subset of C Library functions, selected from the list published in the *SAA Common Programming Interface C Reference - Level 2* as most applicable to TPF applications.

ISO-C C Language Support

In addition to the benefits provided by C/370 support, ISO-C C language support provides the following:

- Standard C functions
- ISO-C library interfaces
- Capability for multiple C libraries to be loaded to the BSS
- Offline tools:
  - C load module build tool
  - library interface tool
  - C stub generator
  - TPF ISO-C startup code.

Communication Support

The TPF 4.1 system connects to the latest level IBM Systems Network Architecture (SNA) communication networks. TPF SNA networks are owned by an MVS/VTAM communication management configuration (CMC) that is attached to and owns the resources in the TPF network. The TPF system maintains all logical unit sessions with the elements in the network. In the event of a CMC failure, the existing sessions with the network are not disrupted. New sessions are established when the CMC has been recovered.

If the TPF system fails, the network remains active. The TPF system usually can be brought back online in time to prevent a loss of the network, therefore avoiding the lengthy restart procedures that are associated with very large networks.

TPF Advanced Program-to-Program Communications (TPF/APPC) is an interface that allows TPF transaction programs to communicate with remote SNA nodes that have implemented the APPC interface using LU 6.2 protocols. TPF/APPC supports all APPC base functions as well as some optional functions.

The TPF 4.1 system also supports communication with other host systems. The TPF system can communicate with other TPF, IMS, CICS, or DATABASE 2 (DB2) systems through channel-to-channel (CTC) connections and the Network Control Program (NCP).

Product Support Function

An MVS operating system controls some functions in a TPF system in a supporting environment to maximize the performance of the TPF production system.

The product support functions are:

- Building and maintaining the applications that are run on the TPF production system
- Maintaining the online operations such as performance analysis and database maintenance
- Processing batch applications using TPF system data such as management reports

TPF System Features

The TPF system has the following features:

- TPF Application Requester (TPFAR)
- High Performance Option (HPO)
- Multi-Processor Interconnect Facility (MPIF)
- Softcopy Publications.

Note: The IBM C/370 language support, which was a feature with the TPF 3.1 system, is included in the base product of the TPF 4.1 system.

TPF Application Requester Feature

The TPF Application Requester (TPFAR) feature allows TPF application programs to read and write data directly to DATABASE 2 (DB2) Version 2 Release 3 or later. The TPFAR feature supports IBM's System Application Architecture (SAA) strategy for database distribution using the remote unit of work method of distributed access. Distributed access involves working with data that is located on remote systems.

The remote unit of work method involves Structured Query Language (SQL) commands imbedded in a local host program used to access a remote relational database (RDB) manager.

The TPFAR feature implements the requester portion of the Distributed Relational Database Architecture (DRDA), allowing TPF application programmers to send static SQL requests to a remote DB2.

Figure 9 shows an SQL request using the TPFAR feature. When the TPF system application makes an SQL request, the TPFAR feature, using TPF Advanced Program-to-Program Communications (TPF/APPC), forwards the request to the application server (AS) on DB2. All communications protocols are handled by the TPFAR feature. DB2 then completes the requested work and returns the result to the TPFAR feature. TPFAR returns the result to the application. TPFAR can also be used to move data from DB2 to the TPF system.

To use the TPFAR feature on your system, you need the following:
- TPF Application Requester feature
- DB2 Version 2 Release 3 or later
  - ACF/VTAM Version 3 Release 3 or later
  - MVS Version 2 Release 2 or later.

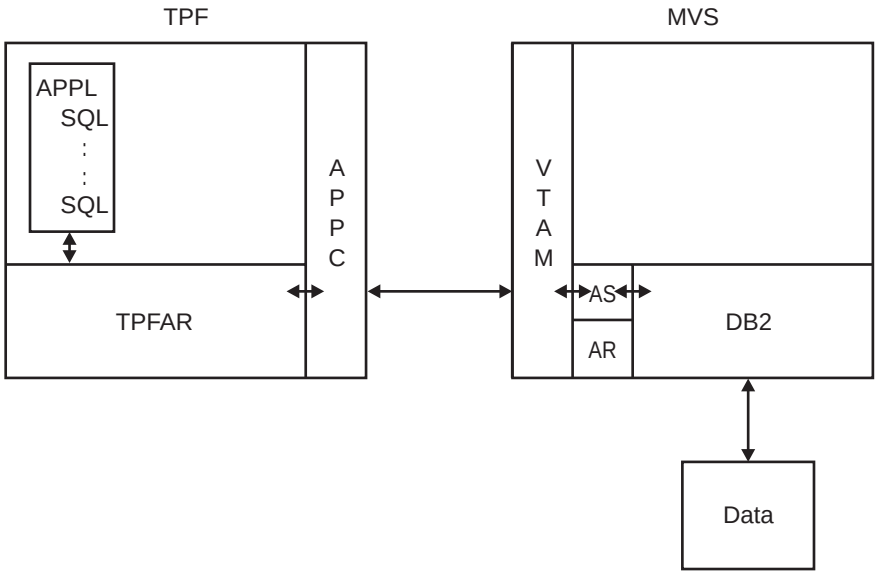

Figure 9. SQL Request Using the TPFAR Feature

High Performance Option Feature

The High Performance Option (HPO) feature, consisting of the loosely coupled (LC) facility and the multiple database function (MDBF), allows the TPF system to run in a loosely coupled configuration where each CPC can share a common database. The High Performance Option feature provides two distinct sub-functions:

- Loosely coupled (LC) facility allows the TPF system to run in a loosely coupled configuration where multiple CPCs can share a common database.
- Multiple database function (MDBF) provides the user of TPF with the ability to separate multiple databases physically or logically.

The Multi-Processor Interconnect Facility (MPIF) feature is a required feature for the High Performance Option (HPO) feature.

Multi-Processor Interconnect Facility Feature

The Multi-Processor Interconnect Facility (MPIF) feature permits TPF CPCs to communicate with each other at very high speed.

Softcopy Publications Feature

TPF publications will be available in softcopy at the general availability of the TPF 4.1 system.

TPF Technical Concepts

This chapter provides a brief overview of TPF processing facilities and concepts. For more information, see *TPF Concepts and Structures*.

Processor Control

A TPF processing complex is characterized by real-time inquiry and update of a large, centralized database with extremely fast response times.

The TPF system:

- Controls incoming and outgoing messages over a variety of communication links
- Assigns and releases main storage and file storage as requested by application programs
- Maintains queues of entries waiting for event completion or more processing
- Reactivates work in progress on a system priority basis
- Services all input/output operations
- Identifies, logs, and resolves (where possible) all equipment errors
- Communicates with the system operator, providing information considered necessary by the control system or requested by the operator
- Provides an operating environment for customer applications.

Entry Management

In a TPF system, a unit of work is generally started by a message received from the user of a terminal or workstation or another processor. When a TPF system receives a message, it assigns a storage block called an *entry control block (ECB)* to the message. The term *entry* refers to the processing associated with an ECB. An entry is the unit of work in the TPF system The life of an entry is measured from the creation to the deletion of the ECB. Programs that require an ECB are called E-type or ECB-controlled programs.

The TPF system is an interrupt-driven system. Interrupts occur as a result of the receipt of a message, the completion of a file request, and the processing of a supervisor call macro. When an entry completes processing, or when an entry requests a service that requires that entry to wait, a TPF control program finds the next entry for processing.

The TPF system uses lists to schedule and dispatch entries. Entries are processed in a priority sequence that allows those entries in progress to complete before new entries are started. The fast response time to individual entries is achieved in part by this technique.

Storage Management

The TPF system divides storage into three classes:

- ***Main Storage***

  This is program-addressable storage in which instructions and data can be stored. A Virtual File Access (VFA) facility allows a user-defined set of file storage records and programs to reside in main storage for performance purposes.
- ***File Storage***

This storage is maintained on direct access storage devices (DASDs). It is used to hold programs and data that must be readily available, such as backup data for system recovery.

- ***Auxiliary Storage***

  This storage is used to hold data and programs that have limited retrieval requirements during the life of an entry. Such data is processed at a noncritical time, usually in a system other than the TPF system. The devices used for auxiliary storage are tapes and disk files.

System Restart

If downtime occurs, the TPF system restarts automatically or by an operator message. A copy of the fixed area of main storage, a restart program, and an IPL program are kept on file storage for restarting the system.

TPF hardware systems are usually duplicated to reduce exposure to equipment failure. The switching of equipment, other than CPCs, can involve physical switching, an operator action, or a restart. The switching of CPCs is done by the operator who manually restarts the other CPC. Backup procedures for a system with HPO are slightly different. With HPO, the number of CPCs are expanded or contracted as required by system load. Backup occurs by contracting the complex and then switching, followed by expanding the complex. TPF preserves critical application data across a downtime. This is possible because the file copy of the critical data located in main storage is updated periodically.

Communication Support in an SNA Environment

Figure 10 on page 17 shows TPF communication support through multiple Systems Network Architecture (SNA) computer networks. The TPF system allows the sharing of lines, terminals, and applications in an integrated network of computing systems. An ACF/VTAM communications management configuration (CMC), which resides in a central processor complex (CPC) other than the TPF system, is necessary for all SNA communication. The CMC and its associated network management products provide services such as resource and configuration ownership and management, session establishment, problem determination, and operational management for the resources in the network.

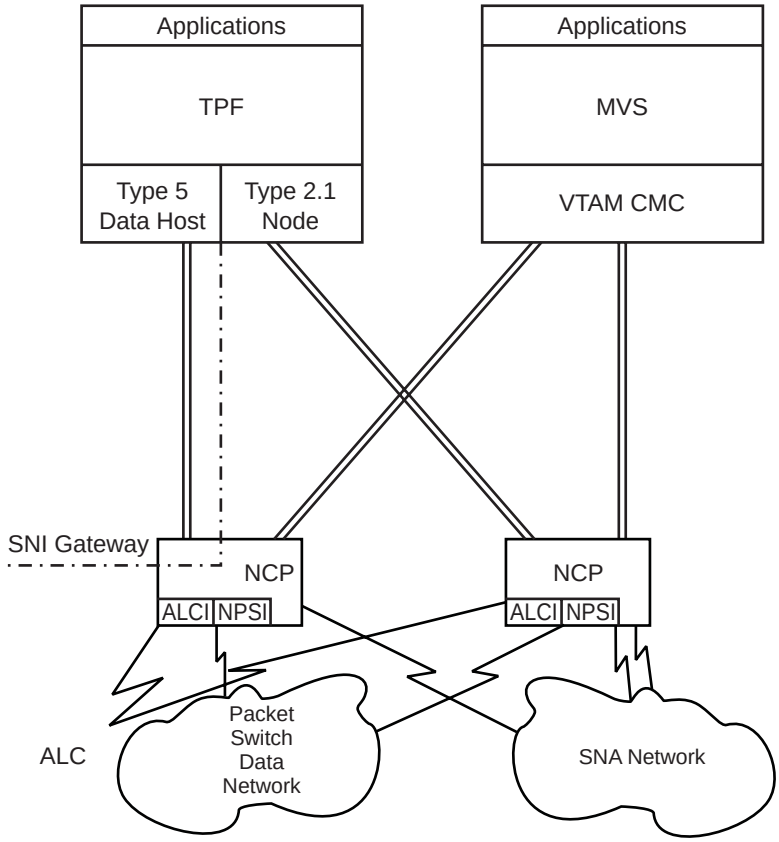

Figure 10. TPF Communication Support

The TPF system supports the Airlines Line Control Interconnection (ALCI) feature of the Network Control Program (NCP) to extend SNA benefits to Airlines Line Control (ALC) communication networks. This extends SNA's inherent advantages of function, connectivity, and network management capability to ALC users. The TPF system's support of the ALCI feature of NCP permits ALC lines to interface to the SNA network, allowing greater use of SNA networking facilities and the potential for controlling line costs through consolidation. The TPF system can attach to an SNA network as either a type 2.1 node or a type 5 data host. The TPF system attaches to the SNA network by means of 3745 communication controllers running the currently supported level of the Network Control Program (NCP). The TPF system can take advantage of the network management capabilities of VTAM. The ACF/VTAM application facility, Logon Manager, provides support for establishing a TPF session as a type 2.1 node. As a type 5 data host, the TPF system must be in an adjacent SNA network, communicating through the services of SNA Network Interconnection (SNI) gateways provided by the channel-attached NCPs. The ability of the TPF system to support both connections to the network allows users to move the TPF system to a type 2.1 node without disrupting their existing network. The TPF system supports cross-system message routing. Data can be transmitted across TPF CPCs without communications management configuration (CMC) intervention after a session is established.

TPF sessions between two logical units are not interrupted by the restart of a failed CMC. Ownership changes resulting from the restart are reflected in the TPF system without a network generation.

The SNA Network Interconnection (SNI) facilities allow the TPF system to connect to a full function SNA network. SNI permits TPF type 5 data host users to take advantage of many advanced SNA networking facilities that are not supported by the TPF system as a data host. By separating TPF CPCs through an NCP gateway from the VTAM owner and network resources, facilities such as transmission groups, alternate paths, transmission priorities, and flow control can be used to improve the level of function and availability in the network.

Users of the TPF system as a type 2.1 node can take advantage of the same level of network function without SNI being required in the local NCPs.

The TPF system supports addressability via SNA Extended Network Addressing. This addressing provides selection of resources from a maximum of 255 NCPs, each having 64KB resources, to a maximum of 16-MB resources in the TPF network. Of importance to the system user is the ability to maintain NCP and VTAM currency and release consistency across the entire network.

The TPF system as a type 2.1 node supports the multiple channel attachment abilities of the 3745 communication controller. This allows one 3745 communication controller to be shared by several processors. A 3745 can support up to 16 host systems, and Enterprise Systems Connection (ESCON) supports up to 64 host systems.

Communication Between Non-SNA and SNA Facilities

Figure 11 on page 19 shows communication between the following distinct TPF system communication configurations:

- Communication between non-SNA facilities is valid only for TPF-to-TPF system connections and requires a user-written relay program in both processors. Communication protocols supported are BSC on 3745 Communication Controllers and Synchronous Link Control (SLC).
- Lines controlled by the EP require a user-written relay program to communicate between non-SNA and SNA facilities. The communication link between the NCPs is SDLC, an X.25 link, or a token ring.
- Communication between multiple SNA networks is through any supported SNA protocol. In this example, the MVS/ESA system contains an IMS or a CICS application that communicates with TPF applications.

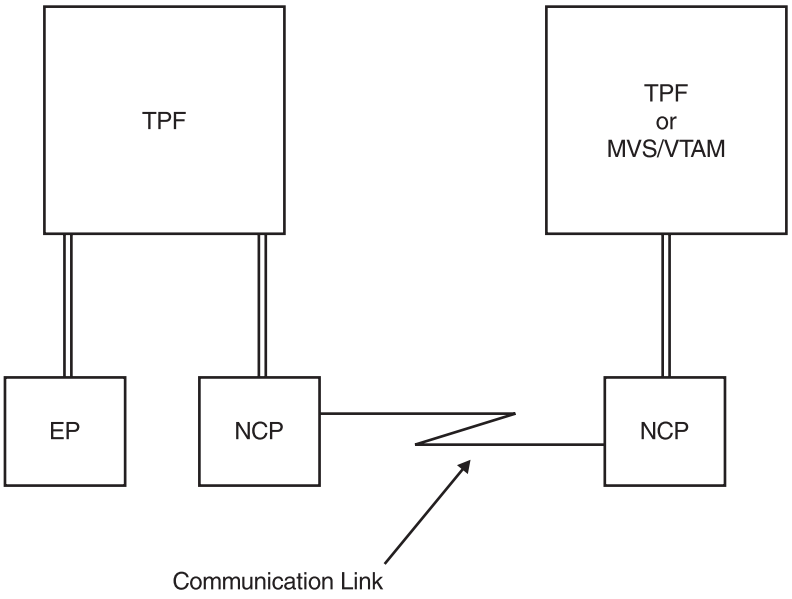

Figure 11. Communication Between Non-SNA and SNA Facilities

Communication Integrity

In a high-performance telecommunications system, the time required to carry out a restart after a system interruption must be short. To keep restart time to a minimum, the TPF system does not automatically reinitialize the network on restart. The host and cluster controllers preserve message sequence numbers over a system interruption, and no network initialization is required. Error conditions such as NCP slowdown, automatic network shutdown, and lost input or output messages are handled as exception conditions by error recovery routines.

Support of inbound and outbound message traffic pacing between application programs or supported terminals helps avoid buffer overrun.

Dynamic collection of tuning statistics support is also provided. I/O interface data can be dynamically collected. Such data can aid a user in selecting the best values for system parameters.

The TPF system fully supports the message integrity capability provided in SNA by message sequence numbering. As a system generation option, the TPF system provides a message recovery package to ensure that each input message is processed successfully and that its reply is delivered successfully. The level of recovery is user-selected, and it extends from full recovery of all input and output messages to simply keeping track of each input message being processed. The TPF system will maintain the message security over a system interruption. Message security and integrity facilities differ according to line protocol and remote devices.

The TPF system also provides an application-oriented recovery package that allows message recovery and database recovery to be combined. In addition, the TPF system provides support to encrypt and decrypt messages to or from any device.

Operator Commands

The primary responsibility for control of the SNA network lies with the VTAM CMC. The TPF operator has the ability to control and monitor TPF system attachment to the network through operator commands. The status of an NCP, a cross-domain resource manager (CDRM), or an individual logical unit can be displayed at the operator console.

Network Environment

The TPF system supports several communication protocols and devices that can be used in combination. TPF systems support very large networks and provide extremely fast response time. This support includes the sharing of terminals among CPCs in an integrated network of computing systems that may run the TPF system or VTAM operating environments. The terminals or workstations in a TPF network can be configured in two ways:

- Direct channel attachment to the processor in which TPF operates (local terminal)
- Connection to a communication controller that is attached to the TPF processor by a channel or a communications link to a local communication controller (remote terminal).

Local Terminals

Local terminals are directly attached to a multiplexer channel on the TPF processor. The principal uses for a local terminal are:

- Primary TPF system console
- Alternate TPF system console
- Terminals used for network control and application program testing.

Remote Terminals

Communication with remote terminals takes place via the IBM 37x5 communication controller. The TPF system supports several 37x5 communication controllers. For additional information, see the *TPF Migration Guide: Program Update Tapes* (available prior to the availability of the TPF 4.1 system).

Performance Considerations

The performance of the TPF system must be monitored to organize system resources properly for peak operating efficiency. The TPF system provides data collection and data reduction programs to measure system performance. These programs provide operational data on activities such as milliseconds CPU busy per message, DASD accesses per message, memory usage per message, program calls per message, message rate, and message length. With this information and the transaction history, you can determine how efficiently the system is running, where bottlenecks occur, and what changes can improve system performance.

Data collection and data reduction provide:

- A tool that can be used during the installation and post-cutover phase to tune the system to peak efficiency
- A means to bring about regular monitoring of system performance
- Statistics for observing long-term trends to provide a base for predicting the growth and needs of the system.

Data Collection

Data collection can be run in continuous mode or sampling mode, allowing multiple types of data to be captured while avoiding significant interference with message processing. All data collection programs write the captured data to an online tape. No attempt is made to analyze the data online, as this would have a negative impact on the system that is being measured.

The three basic techniques used for collecting data are:

- Reading out counters that are embedded in and updated by the TPF system
- Intercepting specific events, such as database macros and program entries when those collector programs are active
- Sampling parameters that fluctuate with time, such as I/O device queues and memory blocks in use.

Data Reduction

All data reduction is performed on an MVS system. The data reduction reports are intended for use by an analyst familiar with the TPF system. Frequency distribution reports including means, standard deviations, and variances of many parameters are available.

The aim of the initial analysis phase of a working system is to establish the normal limits for each of the key factors affecting performance. Once these limits are set and agreed to be realistic, a periodic system check becomes routine.

The analysis of performance data must always start with summary reports. These reports provide key data required for history and trend analysis. When investigating a problem area, the more detailed plot reports or the specialized reports of the DASD and message reduction programs are used. The plot reports, which show the value of each parameter sample in chronological order, are very effective for analyzing the cause-and-effect relationship between parameters.

TPF 4.1 System Enhancements

This chapter describes the following major TPF 4.1 system enhancements.

Transaction Protection and Data Integrity

The TPF 4.1 system separates and isolates information into types of address spaces. The two major address spaces are for system processing and message processing. In the TPF 4.1 system, each message has its own address space.

Through the use of the dynamic address translation (DAT) facility and low address protection, the TPF 4.1 system changes how storage is physically and logically used for system programs, application programs, and messages.

The DAT facility provides a virtual storage environment for the running program and detects whether the program is storing into address spaces other than its own. Each message has private storage, both above and below 16-MB line that is not accessible by other transactions. However, message information that needs to be shared can be shared by other messages through common storage below 16-MB. The TPF system uses the DAT facility to provide each message with private areas that cannot be damaged by other messages or by the TPF system when it operates on behalf of other messages.

This program isolation prevents some of the storage-sharing techniques used in previous releases. However, messages can still share common information in storage through a pool of working storage below 16-MB called the common area.

The low address protection facility protects the first 512 bytes of storage against any alteration by an application program or the TPF system regardless of the storage key used.

Increased Main Storage for Application Use

Using the TPF 4.1 system, your application benefits from increased access to storage above and below 16-MB, while maintaining the 24-bit application program interface (API) for existing Transaction Processing Facility system software. The concept of virtual storage replaces the concept of real (main) storage.

The TPF system supports up to 2GB of main storage. With the use of the dynamic address translation (DAT) facility, TPF system programs can view and manipulate storage located above 16-MB as if the storage were below 16-MB, removing the 16-MB constraint and allowing program access of up to 2GB of main storage. Each message can use up to 2MB of memory for its private use and then reference any other data that is available in main storage.

The TPF system's use of storage is streamlined and more storage is available below 16-MB for application programs. Programs written using 31-bit addressing can use a separate, exclusive 31-bit program area above 16-MB. All large TPF system tables are supported above 16-MB.

Improved TPF System Availability

By using the TPF 4.1 system, your system needs less downtime for software maintenance because your installation can dynamically:

- Change storage allocation values and then reinitialize the TPF system to start the values
- Add SNA terminals without stopping the network
- Assign new programs online without reinitializing the TPF system to activate the programs
- Change program attributes without reinitializing the TPF system to activate the attributes.

E-Type Loader Enhancements

The E-type loader loads programs while the TPF system is in production. The TPF 4.1 system solves the interface problem that occurred during E-type loading in the TPF 3.1 system when there was a window of time in which both old and new versions of a program could be used.

E-type loader enhancements allow easier grouping, allocation, loading, manipulation, and display of programs and loadsets. A loadset groups programs under a unique name so that the E-type loader can perform functions on all the programs at the same time. In TPF 3.1 systems, only 25 programs could be loaded at a time. However, in TPF 4.1 systems, your installation can determine the number of loadsets and the number of programs in the loadsets. A rules database also allows your installation to tailor the way the E-type loader processes under certain circumstances.

Auxiliary Loader

The auxiliary loader, also referred to as multiple TPF images, allows your installation to define up to 8 images of the TPF system on a single processor. Maintaining multiple and separate TPF images allows your installation to integrate program changes more easily by:

- Permitting you to perform loads while the TPF system processes messages without destroying the existing program base
- Providing the ability to fall back immediately to a previous program base without reloading the previous program versions.

Simplified System and Program Allocation

Your system allocation process is simplified using the TPF 4.1 system. The system allocator program (SALO) compiles, link-edits, and runs in one job. Before TPF 4.1 system, allocation was staged in several jobs.

Your installation can change the characteristics of program allocations online. Beginning with this version, most of the system generation information required to run a program is merged in the allocation process. The program allocation options are simplified.

ISO-C C Language Support

Implementing ISO-C C language support on TPF can reduce the cost of developing and maintaining application programs on TPF.

- Many new programmers are trained in C language, reducing the need for extensive in-house training. Also, ISO-C support reduces the amount of TPF-specific knowledge that a programmer needs in order to be effective.

- Having programmers trained in a standard programming language using the same API permits your organization to move programmers across platforms as your business needs change.
- Writing code in a standard programming language greatly increases the possibility of code re-use.
- Application programmers have a set of tools and procedures for building the new program objects. For debugging purposes, TPF dumps and traces are changed to present storage formatted so that it is easily used by both application and coverage programmers.
- System programmers accrue all the benefits of application programmers, plus they are provided with tools to build C load modules and C function libraries. Once ISO-C support is installed, trace information and storage tools are available to help the system programmer.
- Programmers who support application development tools and procedures can appreciate that multiple source programs are link edited together into large load modules.
- When properly tuned, applications written in ISO-C have shorter path lengths than TARGET(TPF) segments.

Greater than 4KB Program Support

ISO-C C language support creates programs in load module format. The C load modules are not restricted to 4KB in size. Also, all C load modules run in 31-bit addressing and are treated as core resident programs.

The C load module build tool is provided to help automate the process of building C load modules.

General Loader Enhancements

User productivity and system management are improved by allowing you to have more control over loading programs and by removing system allocation restrictions. In the TPF 4.1 system, the number of programs that can be allocated has increased from fewer than 33 000 to more than 1 000 000 programs, which improves system management.

Expanded File Addressing Capacity

The TPF 4.1 system supports two new file addressing formats: FARF4 and FARF5. FARF4 is a migration step between the present TPF FARF3 address scheme, and increases your system's database addressing capacity from 640 million to 1 billion 2^{30} records. Migration to FARF5 further expands your addressing capacity to 4 billion 2^{32} records. In the TPF 4.1 system, FARF3 addresses are still supported, but cannot coexist with the new FARF5 addresses. As a result, you must migrate to FARF4 before proceeding to FARF5.

TPF Advanced Program-to-Program Communications Enhancements

The TPF 4.1 system incorporates additional base APPC functions to TPF Advanced Program-to-Program Communications (TPF/APPC) support to complete TPF system support of all APPC base functions. It also adds some of the optional functions defined by the LU 6.2 architecture such as mapped conversations (for the C language interface only). The maximum number of parallel sessions between the

TPF LUs and the remote LU 6.2 nodes increases to an architectural limit of more than 8 million LU sessions and 8 million LU 6.2 sessions.

SNA Communications Enhancements

In the TPF 4.1 system your installation can install new network definitions without disrupting the network or recycling processors. In addition, your installation can write network definitions to tape or general data sets. The definitions become shared between processors during processing, allowing your installation to perform either fresh or dynamic loads from any processor in a loosely coupled complex. If problems are found with the network definitions, your installation can fall back to the previous network definitions, regardless of how many CPUs may have incorporated the new resource definitions.

Systems automatically establish LU–LU sessions over type 2.1 network connections when needed. See “Diagnostic Tools” for a discussion of the new communication traces.

Diagnostic Tools

The following enhancements have been made to the diagnostic capabilities in the TPF 4.1 system:

Branch trace facility
Provides a list of the last 1000 executed BALR, BASK, and BASSM branches using either the ESA/370 or ESA/390 architecture.

Real time trace (RTT)
Assists in determining the causes of errors by selectively monitoring the execution of ECB-controlled programs and producing a record of the system services macros executed and related data.

Online mini dump
You can activate a dump data user exit to present ECB-controlled SERRC and SNAPC data.

User exits provide a means for users of TPF to add user-unique processing at various points in TPF programs without having to modify the released programs.

Macro trace
Two traces are provided:

- Specific to an ECB
- A collated trace, by time stamp, produced for all the ECBs on each processor.

System log trace
Traces events formerly included in macro trace that are not associated with a message or an I/O device.

Enter/Back trace
Traces all ENT*x*C and BACKC macros.

I/O trace
Improved to trace each device separately rather than as a single table for the entire system, allowing your installation to detect devices that are hung. The trace information is collated by time stamp.

Register trace
Traces registers on every supervisor call instruction (SVC).

Program Event Recording (PER) facility

Provides a hardware trace to trap program information when a specific event occurs. The TPF 4.1 system supports the following PER functions:

- Storage alteration
- Instruction fetching
- Successful branching (only on ESA/390 processors).

Path information unit (PIU) trace facility

Provides expanded PIU tracing with additional information.

SNA I/O trace facility

Provides a detailed I/O trace for SNA I/O interrupts that occur during Network Control Program (NCP) XID exchanges, adjacent link station (ALS) XID exchanges, and channel-to-channel (CTC) XID exchanges. It also provides a detailed I/O trace for NCP, ALS, and CTC asynchronous interrupts.

System Error Dump Enhancements

Your installation can control the content of a TPF dump, which is divided into two sections:

- The processor status, trace tables, and virtual memory for message address spaces
- System storage areas.

System error options are no longer subsystem unique. Your installation can define multiple sets of system error numbers and specify additional main storage areas to be dumped. Dumps can also span multiple tape volumes.

Enhanced Tape Support

The TPF 4.1 system provides:

- Automatic tape mounting
- System detection of lost tape interrupts and stalled tape module queues.

Also, beginning in the TPF 4.1 system, the TPF system supports only 3480 tape and tape control devices.

Your installation can do the following without operator intervention:

- Mount alternate tapes on write-enabled devices.
- Convert alternate tapes to active tapes for tape macro processing.
- Convert alternate tapes to standby tapes for tape switching. Automatically mounting alternate tapes improves tape switching during the dump process, enabling your installation to minimize the number of tape devices used by a multiple volume tape dump.

The TPF system detects and reports long or lost tape interrupts and stalled module queue conditions.

Migration Aids

The TPF 4.1 system provides the following migration aids to protect your investment:

- *TPF Migration Guide: Program Update Tapes* (available prior to the general availability of the TPF 4.1 system)

  Explains the TPF 4.1 system details as they affect your system and application code.
- File address reference format (FARF) migration path

  Provides file address reference format 4 (FARF4) as a transition step between FARF3 and FARF5. For more information, see “Expanded File Addressing Capacity” on page 25.
- Virtual-equals-real (VEQR) operating mode

  Provides limited virtual function to help your installation convert from a non-virtual to a virtual TPF system. VEQR mode allows your installation to run programs that are unchanged from the TPF 3.1 system even though the programs use data-sharing techniques that are no longer supported. When unsupported storage sharing is found, the TPF system logs the incident. Running in VEQR mode identifies illegal storage references between address spaces. By using VEQR mode in a test environment, you can test individual programs as you make changes, before modifying your entire application for the TPF 4.1 system. VEQR mode allows you to migrate your applications to the TPF 4.1 system gradually.
- Block checking mode

  Marks code that uses block storage management practices that could have worked in the TPF 3.1 system but do not work in the TPF 4.1 system. Block checking mode marks coding errors such as writing beyond the end of a block, passing blocks chained to other blocks, and using storage that has already been released. Block checking mode can be turned on and off without reinitializing the TPF system.
- Multiple Transaction Processing Facility images

  Multiple TPF images function as both a migration aid and an auxiliary loader. Being able to define up to 8 images of the TPF system on a single processor gives your installation greater flexibility in migrating to a new release. You can maintain images of both releases and fall back to a previous program base if necessary. See “Auxiliary Loader” on page 24 for more details and for a description of the auxiliary loader.

Virtual File Access (VFA) Enhancements

To improve performance in the TPF 4.1 system, virtual file access (VFA) is always present and active. Your installation can run programs directly from VFA, thereby improving system performance. The online and offline VFA performance monitoring facilities improve your installation’s ability to better tune VFA candidates and VFA resources.

System Service Request Enhancements

Enhancements to the macro decoder and supervisor call instruction (SVC) definitions increase an application’s ability to request system services.

The primary interfaces for application requests of system services are through macros using the SVC and fast-link macro decoders. (Fast-link macros are macros that do not issue SVCs.) In the TPF 4.1 system, the SVCs allocated for use are increased from 128 to 255, and the SVCs reserved for customer use are increased

from 1 to 32. In addition, two SVC entries (one for IBM and one for customers) can be reserved to support a second-level structure. With this secondary or indexed structure, your installation can define more than 32 000 additional macros.

One hundred fast-link macros are reserved for your application's use. Fast-link macros are more expedient than other macros because they do not issue SVCs that cause system interrupts.

Certain restricted-use TPF system macros now check the authorization level of the requesting program before providing the system service. If a program requests a service that it is not authorized to obtain, the service is not granted. This authorization level is specified as part of a program's allocation information.

Improved Inter-Processor Communication (IPC)

Beginning in TPF 4.1 systems, the only supported path for inter-processor communications is through the Multi-Processor Interconnect Facility (MPIF) feature. MPIF is a required feature for the High Performance Option (HPO) feature.

IBM C Language Support

In the TPF 4.1 system, IBM C language support is no longer a product feature. It is incorporated in the base TPF product to allow system and application growth. TPF 4.1 requires installation of either the IBM C/370 or AD/Cycle C Compiler and Library.

Performance Monitoring Enhancements

Several reports are enhanced to be more usable and present additional information. A new report contains a histogram of storage frame usage.

Improved System Initialization

System initialization is easier in the TPF 4.1 system. The file address compute program (FACE) table generation is handled by a new offline FACE table generator program rather than as part of system initialization process (SIP).

Improved Capture and Restore Operations

The capture and restore utility controls the maximum number of simultaneous captures allowed for each tape control unit, DASD control unit, DASD channel path and tape channel path, based on the channel path activity.

Planning for the TPF System

To install and use the TPF 4.1 system successfully, your installation must install at least the minimum required machine configuration, communication equipment, and appropriate communication lines before system generation. In addition, your installation must have installed the required software for the TPF system and for the TPF features that you need. The following lists provide an overview of the basic tasks for planning TPF system installation and the minimum system requirements. For additional information on software and hardware requirements, see the *TPF Migration Guide: Program Update Tapes* (available prior to the general availability of the TPF 4.1 system).

Basic Tasks for Planning TPF Installation

Perform the following general tasks:

- Train systems analysts, programmers, and operators in the TPF system
- Develop an implementation plan
- Prepare the physical site
- Design and create a database
- Design and create a communication network
- Design terminal and message formats
- Provide the system generation
- Design and implement application programs using TPF macro instructions and software protocols
- Develop procedures to assure adequate data security
- Develop appropriate backup procedures for the application
- Develop conversion or migration procedures and schedules
- Plan for monitoring system performance and tuning the system.

System Requirements

For specific product levels required and detailed information about requirements, see the *TPF Migration Guide: Program Update Tapes* (available prior to the general availability of the TPF 4.1 system).

The basic Transaction Processing Facility system environment requires specific hardware and software. Optional software products and their benefits in a TPF system environment are also listed.

- Required software:
  - The TPF system
    - Appropriate features
    - Application programs written in High Level Assembler or IBM C/370 language
  - MVS/ESA system and associated products
    - IBM High Level Assembler/MVS & VM & VSE
    - PL/1 Optimizing Compiler and Library
    - AD/Cycle C Compiler and Library
    - C/370 Compiler and Library
    - VTAM for SNA communications

    - NCP.
- Hardware:
  – Processor for each system
  – DASD control units and devices
  – Tape control units and drives
  – System console
  – Communications hardware, including local and remote terminals
  – Printers.

  Note: Additionally, RPQs may be required for specific hardware. The *TPF Migration Guide: Program Update Tapes* (available prior to the general availability of the TPF 4.1 system) contains a list of TPF system RPQs.
- Optional software:
  – The TPF Database Facility (TPFDF) licensed program is a database manager for applications that run in a TPF operating environment. The TPFDF licensed program provides:
    - A logical method of database organization
    - A set of standardized assembler macros or C-functions that form the application program interface (API).
  – The Extended Operations Console Facility/2 (EOCF/2) licensed program provides licensed customers of TPF systems:
    - Enhanced console operations, such as the capability for automation
    - The ability to control and monitor multiple TPF host systems from a single workstation in an IBM Operating System/2 (OS/2) environment.

Index

Numerics

A

B

C

D

E

F

G

H

T

V

W

File Number: S370/30XX-20
Program Number: 5748-T14

Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.

GH31-0147-00