UNISYS

Product Information Announcement

o New Release • Revision o Update o New Mail Code

Title

System 80 Model 7E Capabilities Overview

This announces the release of a revision to this document.

The System 80 model 7E is designed to meet your cost/performance needs by using new hardware technology to support the System 80 architecture. It increases processing power while reducing system size, power consumption, and heat generation.

This overview describes the model 7E and all the peripherals it supports. It also provides information on the Operating System/3 (OS/3) software products that are offered in Release 14.

This revision includes additions, corrections, changes, and expanded information pertaining to:

- Hardware configuration
- Peripherals
- Automated system power-on and supervisor loading

To order additional copies of this document:

- United States customers should call Unisys Direct at 1-800-448-1424.
- All other customers should contact their Unisys Subsidiary Librarian.
- Unisys personnel should use the Electronic Literature Ordering (ELO) system.

Announcement only:

Announcement and attachments:
ECC3, MB00, MB01, MBB1, MBB3, SAB, SAE

System: System 80 Model 7E
Release: 14
Date: November 1991
Part Number: 7004 1785-100

System 80
Model 7E

Capabilities Overview

UNISYS

This document is not a contract and does not create any representations or warranties by Unisys. All applicable representations, warranties and covenants are contained only in the applicable agreement signed by the parties.

The information contained herein is subject to change without notice. Revisions may be issued from time to time to advise of changes or additions.

Correspondence regarding this publication should be addressed directly to Unisys Corporation, SPG East Coast Systems Documentation Development, Tredyffrin Plant, 2476 Swedesford Road, P.O. Box 203, Paoli, PA, 19301-0203, U.S.A.

Contents

Introduction

The System 80 model 7E is the first of a new breed of Unisys products that meet your cost/performance needs by using new technology hardware to support the System 80 architecture.

The model 7E system increases processing power while reducing system size, power consumption, and heat generation.

Although small in size, the model 7E is one of the fastest processors in its price range. Offering a full range of programming languages, transaction and data base products, and fourth-generation language (4GL) information systems, the model 7E is a business system backed by our unrivaled maintenance and support reputation. The model 7E represents an industry high in total system integrity and year-round productivity.

It also represents a logical and cost-effective upgrade from earlier System 80 and Series 90 models. There are no penalties in price, performance, migration, or retraining. Benefits of using the model 7E include:

- Continuation of current applications
- Scaled-down power and cooling requirements
- Complete office environment system
- New state-of-the-art peripherals
- Latest OS/3 software

For any data processing application, the System 80 model 7E cuts data processing demands down to size.

Advanced Hardware Features

The System 80 model 7E incorporates a number of advanced hardware features that further enhance system efficiency and user productivity. Among these features are:

- Very large-scale integration (VLSI) circuits that increase system speed and reliability
- TTL-compatible complementary metal-oxide semiconductor (CMOS) technology
- Personal computer system console
- Approximately 250 percent more processor performance than the System 80 model 4

Integrated Interactive Software

OS/3 software gives you a wide range of software products to meet your needs. You can use:

- A variety of programming languages
- Interactive programming aids
- Integrated data communications
- Information and data base facilities
- Interactive communications with the system

System Maintainability

Since the most reliable system can experience an occasional problem, Unisys provides:

- An array of diagnostic facilities to detect any possible software or hardware problem
- Self-monitoring by the system to help predict the possibility of a failure
- Extensive backup and recovery features
- Direct remote maintenance assistance from the Customer Support Center to substantially reduce the number and duration of service calls

Hardware Transition

The primary goals of the System 80 model 7E are:

- Promote migration to the System 80 model 7E by providing an attractive upgrade for System 80 models 3 and 4 users and for Series 90 models 25, 30, and 40 users
- Promote future upgrade from the System 80 model 7E to System 80 models 10 through 20 and model 50
- Provide a departmental/development system solution for System 80 models 8 through 20 users

Series 90 Family and System 80 Models 3 through 6

The highest level of transition support is provided for users upgrading from the Series 90 and the System 80 models 3 through 6. Object program and data compatibility are available to System 80 and Series 90 users who wish to continue operating as they have on previous systems. Object code and data can be transported to the model 7E via tape or communications lines.

Hardware functional compatibility provides full support of communications network devices. Your existing OS/3 UNISCOPE® terminals, PCs with synchronous terminal emulation package (STEP), communications networks, and DCPs are fully supported on the System 80 model 7E system. However, communications programs that use the ICAM CPI interface are not supported on the model 7E, nor are terminals/controllers that do not conform to standard protocol definitions.

Upgrades from System 80 Model 7E

Future upgrade from System 80 model 7E to present and future Unisys systems is an important consideration in the system design. This consideration manifests itself primarily in the following areas:

- High-level compilers
- Data base management support
- Support of Unisys command language standards, including DCA and DDP
- Interconnect services to Unisys U Series systems
- Industry standard SCSI peripheral interface
- Full communications support
- Industry standard 9-track tape support
- Full System 80 architecture

Departmental/Development System for System 80 Models 8 through 20

With the System 80 model 7E, current users of System 80 models 8 through 20 can introduce new applications cost effectively with no impact on their production workload and no need to expand the equipment at their central site. They can move some or all of the processing closer to the end user but still manage the usage and growth of the system from the central site.

System 80 models 8 through 20 users can also use the model 7E for new development, again without impact to the central site. Programmers can do all application development and testing on the model 7E and move the applications to the central system only when completely tested and operational.

Hardware

The basic model 7E processor complex is housed in a single processor cabinet containing:

- Central processing unit (CPU)
- I/O control processor (IOCP)
- Main storage
- Cache memory
- Maintenance hardware
- Up to eight integrated disk drives.
- Data link processors (DLPs)

The DLPs provide I/O support for the various types of peripheral devices supported by the model 7E. The types of DLPs available in the processor cabinet are the DLP for data communications (COMM DLP), the DLP for high-speed standard interface (HSSI) printers, and the DLP for small computer system interface (SCSI).

Two I/O expansion cabinets can be added to the processor complex to provide your system with additional peripheral or communications line capacity. These cabinets provide significant expansion capabilities that let you easily expand your system at your own pace. As your peripheral and communications requirements grow, you can incrementally expand the system to meet your needs.

The first I/O expansion cabinet attaches directly to the processor cabinet. It provides up to eight more DLPs for expanding the peripheral and communications I/O line handling capability of the system. The cabinet also lets you add up to eight additional integrated disk drives.

The second I/O expansion cabinet attaches to the first expansion cabinet and provides the same expansion capabilities as the first.

A comprehensive selection of disks, printers, tapes, and telecommunication devices is available to match your model 7E to your data processing requirements.

Processor Cabinet

The processor cabinet accommodates:

- Processor PCA (printed circuit assembly)
- Processor cache PCA
- DMCU memory control unit PCA
- Memory PCA
- MIC-PC maintenance interface PCA
- TPC power control PCA
- SCP DLP for system console
- I/O channel adapter (IOA) PCA
- SCSI DLP for tape and disk
- HSSI DLP for printers
- COMM DLP for data communications
- Optional (additional) DLPs
- Integrated disk drive

Processor PCA

Two LSI processors are configured on the processor PCA, one used as an OS/3 instruction processor and one used as an I/O control processor.

Processor Cache PCA

The cache available to the processors is 64K bytes each.

DMCU Memory Control Unit PCA

The memory control unit performs memory control for the system. It provides double-bit error detection and single-bit error correction. It also logs soft/hard memory failures and additional maintenance system status.

Memory PCA

Memory is contained in 8-MB modules. A maximum of 16 MB may be configured. All memory configurations are accommodated by a single PCA. Field expansion is accomplished by a PCA exchange.

MIC-PC Maintenance Interface PCA

The maintenance interface PCA and the TPC power control PCA form part of the Personal Workstation^2 (PW^{2TM}) based maintenance subsystem, which performs system initialization, power control, and maintenance.

TPC Power Control PCA

The power control PCA provides power control for the system. It also forms part of the maintenance subsystem and provides the physical connectivity to the system for the PW^2 LAN interface and the RMI interface.

System Console Processor (SCP) DLP

The SCP DLP is a mandatory data link processor that provides the system support for the PW^2 used as the service processor and operator console. It supports one PW^2 system console subsystem.

I/O Channel Adapter (IOA) PCA

Input/output for the model 7E is established by an IOA board that contains a device level interface (DLI) for communicating with the data link processors (DLPs) in the processor cabinet, and two message level interfaces (MLI) for communicating with the DLPs in the I/O expansion cabinets. The adapter runs at 8 MHz and has a data transfer rate of 8 MB/second for the DLI and 5.3 MB/second for the MLI.

Small Computer System Interface (SCSI) DLP

The SCSI DLP connects the host to both integrated and external I/O peripheral devices via the SCSI bus, which has seven addresses. One SCSI DLP is provided with your system. Additional SCSI DLPs can be added at your option.

High-Speed Standard Interface (HSSI) Printer DLP

The HSSI DLP connects the host to an HSSI bus to control the 9246-7T/7TX/14T (650/1200 lpm) impact printers. One HSSI DLP is provided with your system. Additional HSSI DLPs can be added at your option.

Data Communications (COMM DLP)

The COMM DLP connects the host to UNISCOPE terminals, async and bisync lines, TTY™ lines, UDLC/X.25 networks, and the AP9215-1 nonimpact printer. A COMM DLP can support up to four RS-232 interface lines with identical or mixed protocols. One COMM DLP is provided with your system. Additional COMM DLPs can be added at your option.

Additional DLPs

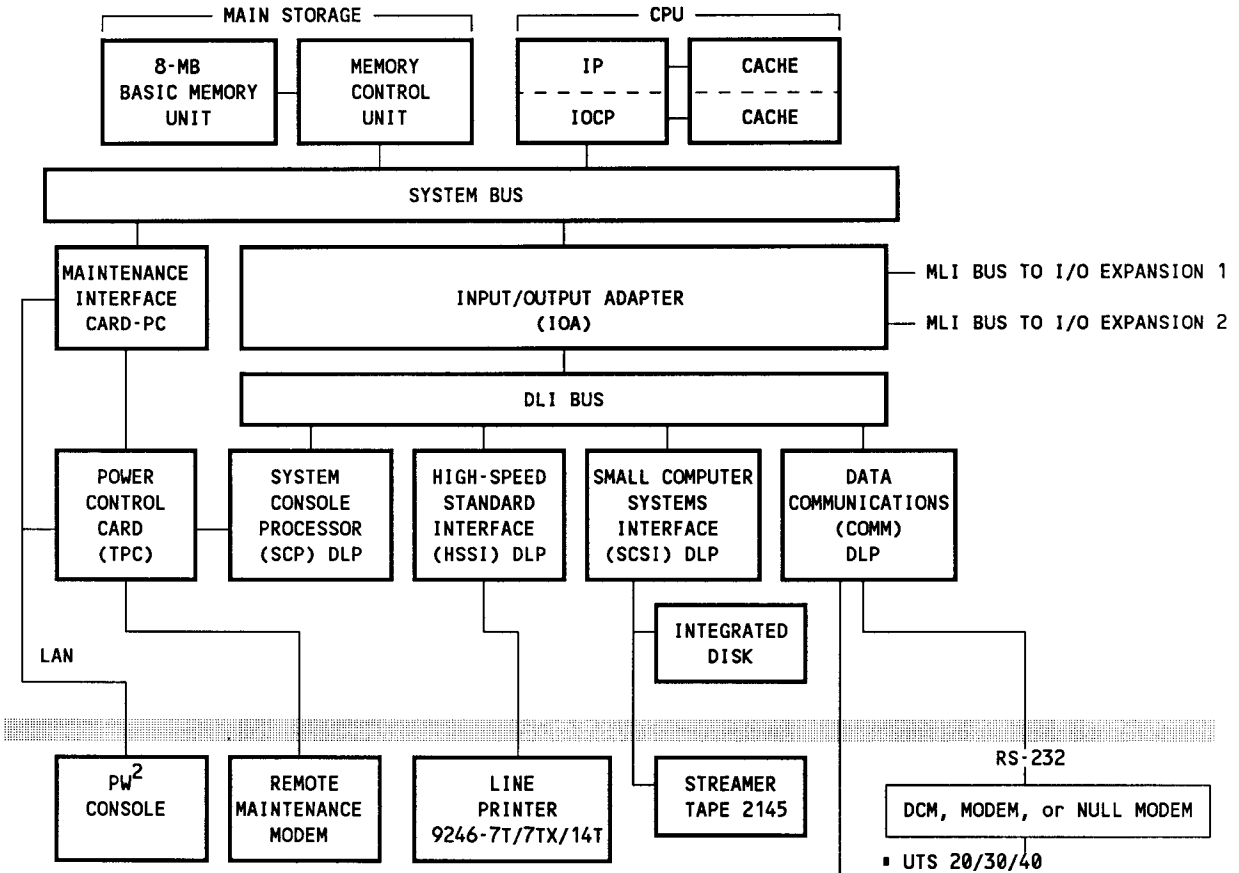
Additional SCSI, COMM, and HSSI DLPs can be added to your system at your option. The processor cabinet can accommodate up to three additional DLPs. Each of the optional expansion cabinets can accommodate up to eight additional DLPs.

Integrated Disk Drive

Up to eight 305-MB formatted disk drives can be housed in the processor cabinet. The disks have an approximate average access time of 26.3 ms, making them among the fastest disk drives offered in the industry today. Up to eight additional drives can be configured in each of the two optional expansion cabinets.

Hardware Configuration

The following diagram illustrates the hardware configuration of a model 7E system. Everything above the shaded line is contained in the processor cabinet. Any additional DLPs would be housed in expansion cabinets (not shown).



SYSTEM MAXIMUMS

All maximums may not be simultaneously achievable.

- Main Storage: 16 MB
- SCP DLP: 1
- SCSI DLP: 6
  - Integrated Disk: 8
  - Streaming Tape: 18
- HSSI DLP: 4
  - Line printer: 4
- COMM DLP: 20
  - Nonimpact Printer: 12
  - Workstations/Terminals/PCs: 76 (Connectivity for workstations, terminals, and PCs can be expanded with the use of TMUXs.)

System Architecture

The system processor and I/O complex consists of these major modules:

- Central processing unit (CPU)
- Main storage
- Input/output (I/O) subsystem
- Data communications subsystem
- System console subsystem

Central Processing Unit (CPU)

The CPU contains the instruction processor (IP), the I/O processor (IOCP), and processor caches. The processors feature high-density CMOS VLSI and advanced packaging to provide high performance and reliability. The IP supports the full OS/3 instruction set and several new features. The IOCP is fully independent and interfaces to OS/3 to control all I/O operations.

Main Storage

The main storage subsystem features 1 MB DRAM chip technology and uses a single memory board. The standard 8-MB memory can be expanded to 16 MB through a simple board exchange to give your system increased speed and power.

The memory board is supported by a memory control unit (MCU), which provides single-bit error correction, double-bit error detection, and processor interfaces.

Input/Output Subsystem

The I/O subsystem consists of an I/O central processor that uses intelligent microprocessor-based peripheral processors called data link processors (DLPs). These microprocessors are designed to service the specialized needs of peripheral devices that operate independently of the central processor.

An I/O expansion cabinet can also accommodate from one to eight additional DLPs and up to eight integrated disk drives. Up to two I/O expansion cabinets may be configured.

The benefit of this subsystem design is a division of the information processing workload, which results in less processor overhead and produces higher total system throughput.

Data Communications Subsystem

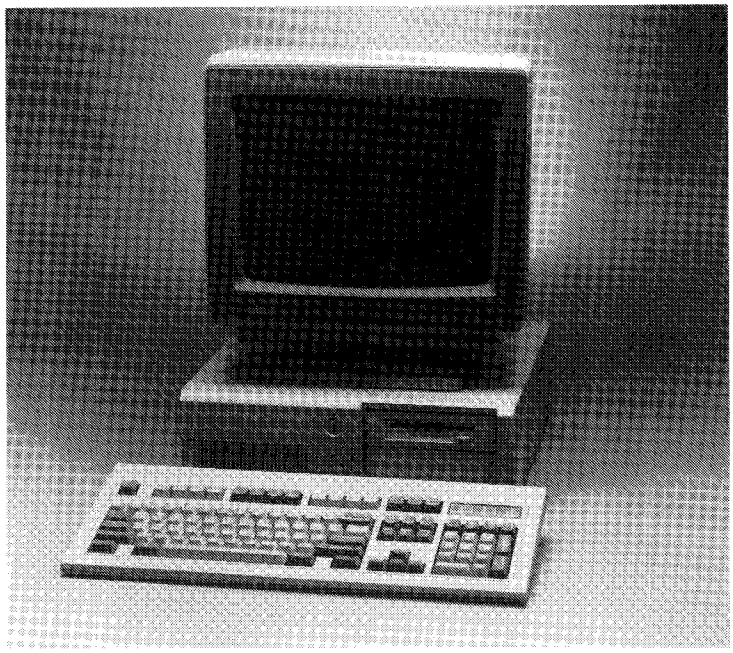
The model 7E uses the data communications (COMM) DLP for data communication interfaces.

This high-performance processor allows the communications protocols to be off-loaded from OS/3 and the CPU. It supports UNISCOPE, UDLC/X.25, bisync, async, and nonimpact printer protocols. Each COMM DLP can provide up to four RS-232 interfaces, which can be configured to support a variety of communications needs.

System Console Subsystem

The system console subsystem uses a Unisys PW^2 personal computer. The subsystem includes an independent maintenance disk, color monitor, mouse pointer, and multitasking operating environment. Operator features include multiple windows and online help. Maintenance features provide system initialization and full-service menus.

The model 7E allows you to use colors to highlight the operator console screen for maximum operator efficiency and screen visibility.



BAL Instruction Set

The BAL instruction set for System 80 model 7E consists of general, floating-point, decimal, and system control instructions. Compatibility with the Series 90 and System 80 systems is maintained on nonprivileged instructions.

The instruction repertoire of the system includes six instruction types:

- RR (Register to Register)
- RX (Register to Indexed Storage)
- RS (Register to Storage)
- SI (Storage and Immediate Operand)
- S (Implied Operand and Storage)
- SS (Storage to Storage)

BAL Nonprivileged Instruction Set

Nonprivileged instructions process fixed-length binary numbers, floating-point numbers, packed and unpacked decimal numbers, and EBCDIC or ASCII characters. Data can be transferred between main storage and the user program set of general registers, as well as from one location in main storage to another. Shifting, branching, logical, and arithmetic operations are also included.

BAL Privileged Instruction Set

Privileged instructions are used exclusively by the operating system software when operating in the supervisory state. This set of instructions includes system control, hardware management, job scheduling, and I/O control facilities to load and store the contents of low-order main storage and to load the writable section of the microprogram control storage. Privileged instructions cannot be included in a user program.

Peripherals

The choices for System 80 model 7E peripheral devices include:

- Integrated 5-1/4-inch SCSI disk drive (305 MB formatted)
- PE/GCR/NRZI SCSI streaming tapes
- Line printers (650 and 1200 lpm)
- Nonimpact printer
- Communications connected terminals with or without attached printers
- DCPs

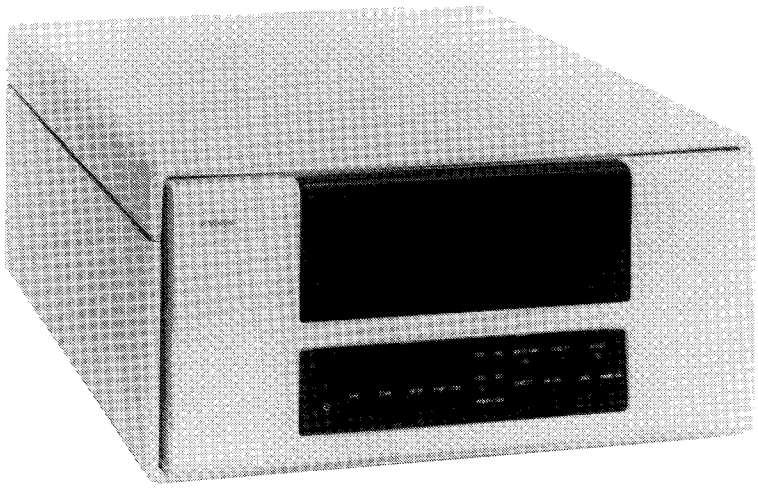
Integrated 5-1/4-inch SCSI Disk Drive

The integrated 5-1/4-inch disk drive uses the SCSI interface. This disk subsystem has a capacity of 380 MB unformatted and a usable formatted capacity of 305 MB. The main processor cabinet can be configured with one to eight integrated drives, providing a maximum storage capacity of 2.4 billion bytes of data within this cabinet. Up to eight additional integrated drives can be configured in each of the optional I/O expansion cabinets available with the model 7E. A system with a full complement of cabinets and integrated disk drives offers a total data capacity of 7.2 billion bytes.

Record format (bytes)	256
Tracks per cylinder	7
Number of cylinders	1773
Sectors/track	96
Data rate	1.6 MB/s
Average access time	26.3 ms
Data capacity (formatted)	305 MB
Disk media	Fixed
Rotational speed	3,600 rpm

2145 Streamer Tape

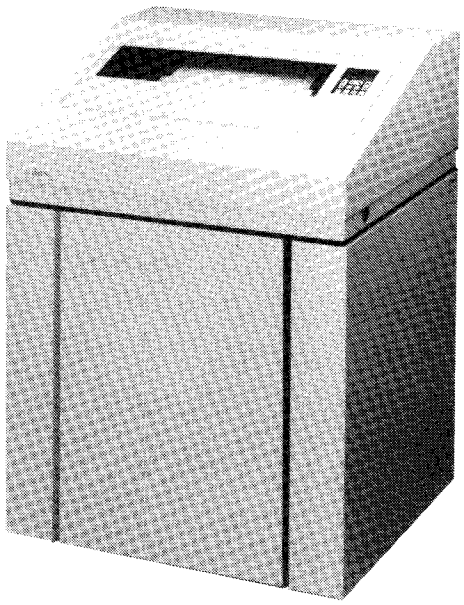
The 2145 streamer tape drive is a high-performance desktop tape drive that uses an open reel half-inch tape and a SCSI interface. It operates at 100 ips in streaming mode and 50 ips in start-stop mode. It is a 270-MB-capacity tape supporting 6250 GCR, 1600 PE, and 800 NRZI. This tape includes features such as read/write error detection and correction, and automatic calibration.



	High Speed	Low Speed
Tape speed	100 ips	50 ips
Rewind time	190 seconds	190 seconds
Data rates	625 KB/s	312.5 KB/s
Data density	800 - 6250 bpi	800 - 6250 bpi
Recording format	9-track PE,GCR,NRZI	9-track PE,GCR,NRZI

9246-7T/7TX Printer

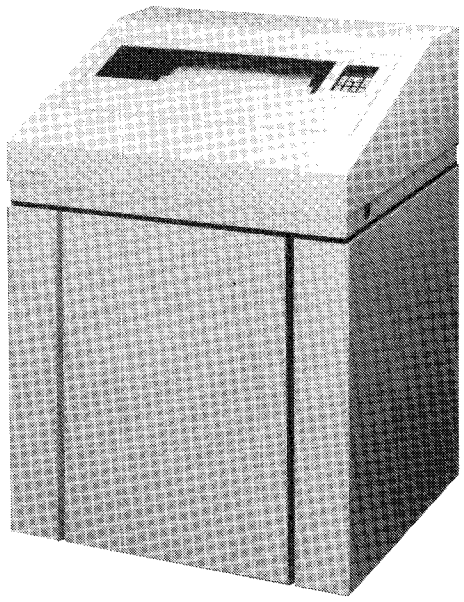
The 9246-7T/7TX printer is a 650-lpm impact printer that interfaces to the HSSI printer DLP and provides high-quality printing. The 9246-7T/7TX provides OCR-readable print quality with dual print speed for enhanced quality. It also prints up to 132 characters/line with operator changeable line spacing of either six or eight lpi. The 9246-7TX printer also includes a power stacker.



Print speed	650 lpm for a 64-character set
Print line	132 columns wide
Printing density	10 columns per inch
Line spacing	6 or 8 lpi (operator selectable)
Line advance	14 ms at 6 lpi
Forms slew rate	15 ips (38 cm/s)
Forms width	3 to 16 inches (7.7 to 41 cm)
Forms length	3 to 14 inches (7.7 to 36 cm)
Number of copies	1 original plus 5 carbons

9246-14T Printer

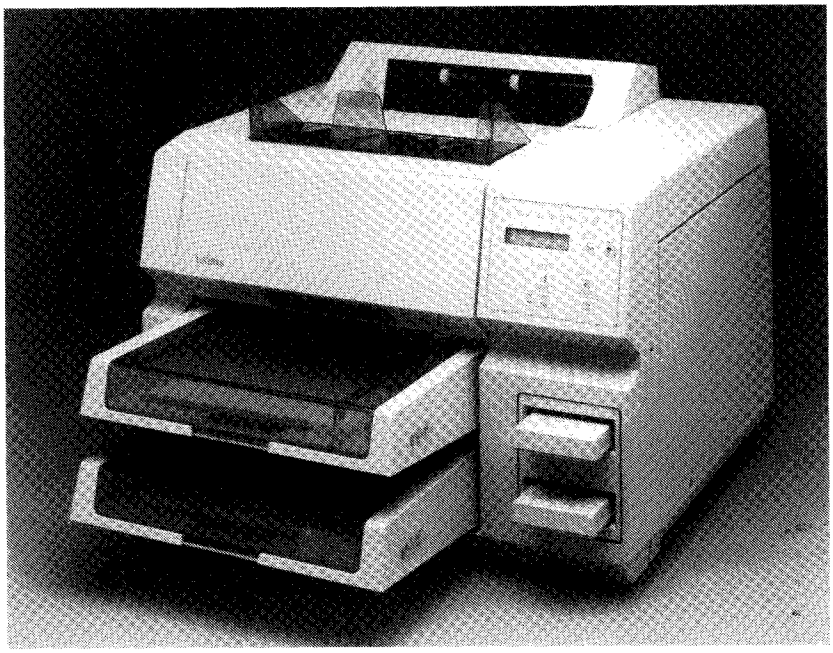
The 9246-14T printer is a moderate-speed band printer that provides quality printing at a rate of up to 1,200 lpm when using a 64-character-set print band. Standard features include power paper stacker, operator changeable print bands, and component testing during power-up.



Print speed	1,200 lpm for a 64-character set
Print line	132 columns wide
Printing density	10 columns per inch
Line spacing	6 or 8 lines per inch (operator selectable)
Line advance	14 ms at 6 lines per inch
Forms slew rate	15 inches per second (38 cm/s)
Forms width	3 to 16 inches (7.7 to 41 cm)
Forms length	3 to 14 inches (7.7 to 36 cm)
Number of copies	1 original plus 5 carbons

AP9215-1 Nonimpact Printer

The AP9215-1 nonimpact printer is a 15-ppm, single-sheet desktop printer with 1.5 MB of user memory. It prints text at 300 x 300 dots-per-inch (dpi) resolution. The Unisys font cartridge program offers flexibility to satisfy almost any printing requirements, such as brochures, parts lists, slides, presentations, scientific and mathematical symbols, scripts, and more. This printer interfaces through the COMM DLP.



Print speed	15 pages per minute
Print line	80, 110, or 132 columns wide
Printing density	10 or 12 columns per inch
Line spacing	6 or 8 lines per inch
Forms width	8.3 to 8.5 inches (21.0 to 21.6 cm)
Forms length	11 to 14 inches (27.9 to 35.6 cm)
Number of copies	1 original

SVT 1120 Terminal

The SVT 1120 terminal is UNISCOPE control code compatible and functions like a UTS 20 terminal but with several significant enhancements.

The monitor is a 14-inch CRT with P31 green phosphor. The operator can select 80 or 132 columns by 25 lines. The display features a CRT saver for time delay display shut-off and a display intensity control. The SVT 1120 connects via a COMM DLP using UNISCOPE protocol.

Supported keyboard-selectable languages include English, British, French, German, Italian, Spanish, Danish/Norwegian, and Swedish/Finnish.

Two menu-driven setup control pages are provided. One establishes operator convenient parameters, and the other establishes communications configuration parameters.

SVT 1123 Terminal

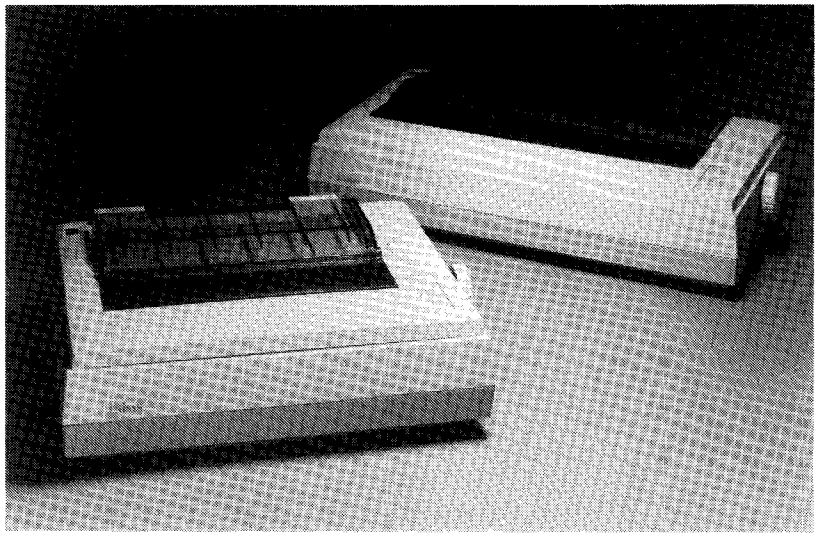
The SVT 1123 terminal provides access to System 80 system response mode and is comparable to the UTS 20. The SVT 1123 is identical in physical appearance to the SVT 1120 terminal. The terminal setup menus include four languages: English, German, French, and Italian.

SVT 1124 Terminal

The SVT 1124 terminal is the same as the SVT 1123 except the four terminal setup languages are English, German, Spanish, and Dutch.

AP1327 and AP1329 Terminal Printers

The AP1327 80-column terminal printer and AP1329 136-column terminal printer are low-cost, medium-speed, impact dot matrix printers that print at a rate of 270 characters per second at 10 characters per inch. The printers include a punch tractor for use with sprocketed paper or continuous forms. They also include a friction feed platen for cut sheets.



Distributed Communications Processors (DCPs)

The model 7E supports DCP models 5/15/25/30/35/40/50/55. The smallest member of the DCP family, the DCP/5 is housed in a Unisys Personal Workstation Series (PW^2). It is ideally suited for small sites where full-function networking is required. Functioning as a remote concentrator, protocol converter, and network interface processor, the DCP/5 can support a maximum of seven lines or five protocols.

Additional Communications Devices

The following devices can also be attached to the model 7E as communications connected devices:

- Unisys PCs (with STEP)
- UNIX systems
- UTS 10/20/30/40/200/400/4000, UVT 1224, SVT 1210/20 terminals
- Remote System 80 hosts
- 2780/3780 devices (BSC terminals and hosts)

Hardware Diagnostics

There are two groups of tests available: system confidence tests, which are provided as part of the system, and customer engineering diagnostics, which are provided to Unisys field service personnel for use in servicing and repairing the model 7E system.

The following are system confidence tests:

- Resident diagnostics - are an integral part of the hardware and perform basic hardware checking automatically when the system is powered on.
- Offline diagnostics - are programs that check the operability of all system instructions. These programs execute under the control of an offline executive.
- Online diagnostics - are programs that check all peripheral and communications devices to ensure proper system function and operation. These programs are under the control of the operating system and run concurrently with user jobs.

The following are customer engineering diagnostics:

- Microdiagnostics - test the central processing unit, cache, memory controller, and I/O adapter.
- Load path diagnostics (LPD) - allow the testing of the SCSI disk from an offline environment.
- Online diagnostics (error log analysis) - log errors in the system error log during system operation. The error log analysis program displays individual entries and summary information. This summary information is also made available for transmission to a remote support center.

Hardware Summary

The following tables summarize the system hardware and communications characteristics of the System 80 model 7E.

System Characteristics

System Orientation	Disk, with Concurrent Interactive and Batch Processing
Central processing unit	Instruction processor (IP)
	Two sets of 16 general-purpose registers
	Four floating-point registers
	I/O control processor (IOCP)
	Priority controller for access to any two units on a priority basis
	Two-stage pipelined processors
	320-KB control storage for each processor
	Maximum system bus transfer rate of 16 MB/s
Main storage processor	32 relocation registers for address conversion (31 for user programs, 1 for operating system use)
	Single-bit error correction, double-bit error detection
Main storage unit	8 MB or 16 MB
IP cache	64K bytes capacity, 250 ns access
IOCP cache	64K bytes capacity, 250 ns access
Block multiplexer functionality via IOCP	Maximum of six channels; each channel can support one to eight data link processors (DLPs)
	One peripheral and one communications channel per cabinet (Processor, I/O Expansion 1, and I/O Expansion 2)

continued

System Characteristics (cont.)

DLP Characteristics	
Data link processors (DLPs)	Four DLP types supported: ■ SCP (system console processor) – One required in processor cabinet for system console. ■ SCSI (small computer systems interface) – One to four per system for: – 1 to 24 integrated disks – 1 to 14 streaming tapes ■ HSSI (high speed standard interface) – One to four per system for up to four 9246-7T/7TX/14T printers ■ COMM (data communications) – One to 20 per system for UNISCOPE, UDLC, bisync, async, and AP9215-1 protocols

Communications Characteristics

Protocol/Function	Line Speed	Mode	Interface
UNISCOPE, UTS 20/30/40, UTS 400/4000, U200, SVT 1120/1123/1124, PC, UNIX	2400 - 19.2K bps	Half-duplex, synchronous	RS-232
BSC 2780/3780 devices	2400 - 9600 bps	Half-duplex, synchronous	RS-232
Async UTS 10/TTY (or equivalent)	150 - 9600 bps	Half-duplex, asynchronous	RS-232
UDLC DCP/5/15/25/30/35/40/ 50/55 System 80 hosts	2400 - 19.2K bps	Full-duplex, synchronous	RS-232
X.25 packet-switched PDN	2400 - 9600 bps	Full-duplex, synchronous	RS-232
AP92 AP9215-1	150 - 19.2K bps	Half-duplex, asynchronous	RS-232

System Platform Software

OS/3 System Platform Software (SPS) directs the efficient and flexible centralized control of System 80 activities. It also gives you a multiple job environment that uses the full capabilities of System 80. This environment allows concurrent operation of programs with immediate response to user inquiries and requests at terminals.

OS/3 is designed to take advantage of the increased speed and capacity of the System 80 model 7E to dramatically improve job throughput. The software makes optimum use of all available hardware facilities, resulting in efficient job turnaround time. OS/3 also supports a wide range of peripheral hardware, which is described earlier in this overview.

SPS consists of the supervisor, job control, and other programs that form the basis of OS/3 operations. SPS also makes the following functions possible:

- Interactive processing
- Communications
- Batch processing
- User program development
- Disk processing
- Resource management

Supervisor

The supervisor provides the central control needed to make the system's hardware, software, user programs, and interactive facilities work together efficiently. The supervisor services make many features possible, including:

- Multiple program processing
- Simultaneous interactive access
- Error control and recovery
- Automatic resource management

The supervisor manages and coordinates all system activities, handles randomly occurring events, initiates and coordinates execution of batch and interactive programs, and provides advanced programming facilities. The supervisor is one of the most complex components in the system, but it makes the System 80 an easy-to-use and efficient data processing system.

Supervisor Functions and Services

Supervisor interrupt requests	Service the various interrupts generated by the system.
Task switcher	Determines the order in which the various tasks are given CPU time.
Physical input/output control	Controls the dispatching, queuing, and interrupt processing for all I/O devices directly connected to the system.
Disk cache facility	Stores data in a reserved cache to improve I/O performance.
Transient management	Schedules, locates, and loads transient (nonresident) supervisor routines.
Timer and day clock services	Provide a system clock for timed, batch, or interactive activities.
System console and terminal management	Controls information to and from the system console and various terminals.
Error recovery and logging	Handles any program or machine error that causes an interrupt.
Multitasking	Concurrently processes more than one task from one or more jobs.
Diagnostic services	Support a variety of diagnostic services to isolate and correct diverse error conditions.
Main storage management	Allocates and deallocates main storage as required by other system control software.
Security	Controls access to the subsystem environment.
Installation verification	Tests functional capability of system components.
Automated system power-on and supervisor loading	Permits an automatic system power-on and supervisor load without operator intervention.

Supervisor Interrupt Requests

A supervisor interrupt stops the normal processor flow and must be completed in some way before processing can continue. Requests are generated by the various hardware and software components. OS/3 recognizes eight types of interrupts.

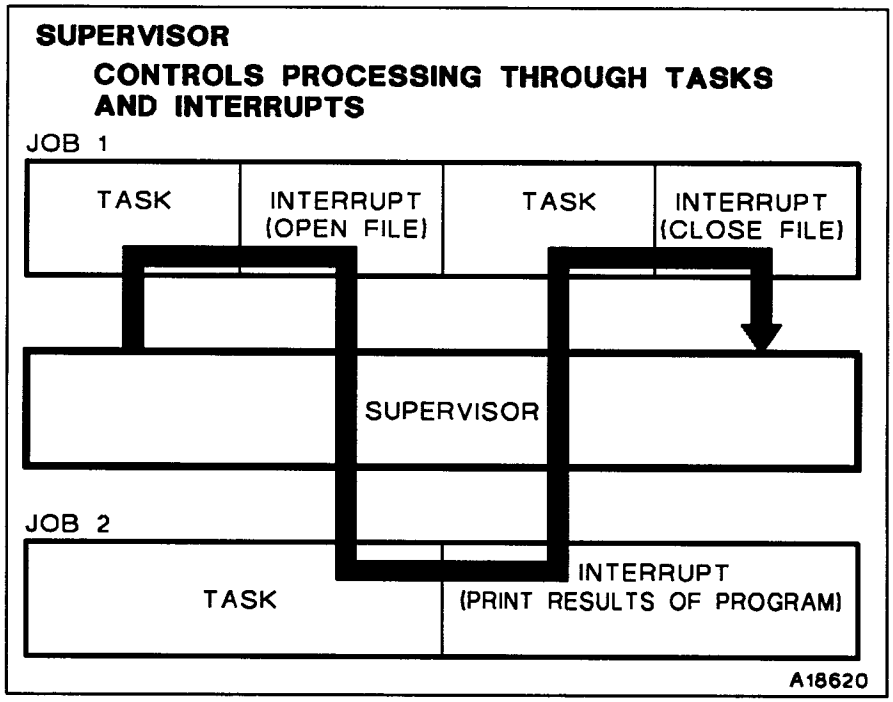
Supervisor Interrupts

Supervisor call	Occurs in response to the SUPERVISOR CALL (SVC) machine instruction. Although it is handled as an interrupt, programs routinely use the supervisor call to request supervisor services.
Input/output	Occurs in response to signals from I/O channels.
Exigent machine check	Indicates a malfunction in or around the processor from which the supervisor cannot recover.
Repressible machine check	Indicates a malfunction in or around the processor from which supervisor recovery is possible.
External interrupt	Generated either by the processor interval time or the system console interrupt key.
Program check	Occurs when the processor attempts to execute a nonexistent instruction or to execute an existing instruction in an illegal manner.
Program event recording	Provides dynamic monitoring of executing programs by storing information about the current instruction when a specified event occurs.
Restart	Occurs when a restart command is issued from the system console and can be used to put a stopped processor in the operating state.

Some interrupts, like supervisor call or input/output, are routine; others, like program or exigent machine checks, represent errors that the supervisor must handle with minimal system interruption.

Task Switcher

A task is the smallest entity to compete for central processing unit (CPU) time. The system, as well as users, can initiate tasks. Tasks initiated by users are interactive activities or functions requested through job control. The system initiates tasks to support user requests or as a part of normal system operation. The task switcher coordinates the processing of all current tasks and decides, based on established priority, which task should be processed next.



Physical Input/Output Control System (PIOCS)

OS/3 performs all I/O operations with peripheral devices through the physical input/output control system (PIOCS). OS/3 PIOCS is composed of software routines that provide maximum throughput on all peripheral devices and allow expansion to support new devices.

PIOCS handles the queuing and initiating of all I/O commands and the processing of I/O interrupts in the following ways:

- Control – PIOCS receives control when a request is issued for an I/O operation. Control is not returned until the I/O request is completed. However, other tasks in the system can be activated if status indicates a ready-to-run condition.

- Queuing – Requests for I/O operations are initially queued, by priority, in device and channel queues.

- Dispatching – Dispatching follows queuing of an I/O order when a path to the device is available.

- Dispatch routines – The I/O dispatch routines perform needed service functions, such as disk address verification and parameter checking. Interrupts from I/O channels are serviced as a high-priority function of the supervisor to free the channel for dispatching other I/O orders that were queued.

- Completion of I/O request – Upon completion of an I/O order, an I/O status analysis is performed to determine whether an abnormal condition occurred.

PIOCS requires operator responses only when unavoidable. However, if specified at SYSGEN, the operating system can respond internally to system messages for the operator.

Disk Cache Facility (DCF)

The DCF increases system performance by reducing the potential for I/O contention to disk devices. A portion of memory is allocated as a disk cache and managed by operating system software. Portions of disk information currently being accessed are transferred to disk cache. Selected I/O functions reference disk cache to determine if the information being requested resides in cache. If present in cache, the requested information is transferred to the requesting program without an I/O request to the disk.

When a particular area on a disk is read, the disk cache facility writes all or a portion of it (depending on the size of the cache entered when DCF was initialized) to a reserved cache storage area. A physical I/O is unnecessary in subsequent reads of that same disk area since that data is already present in storage.

Reducing the number of required I/Os results in the following reductions:

- Queuing delays – I/O requests must wait to be issued in turn.

- Disk seek time – The disk accessor moves to the proper cylinder.

- Latency – The disk rotates so that the disk read/write heads reach the proper record.

- Data transfer time from disk to storage.

Main storage disk cache is supported by software included in the system microcode file. It is called, through the system definition file, to maintain consistency among all System 80 models.

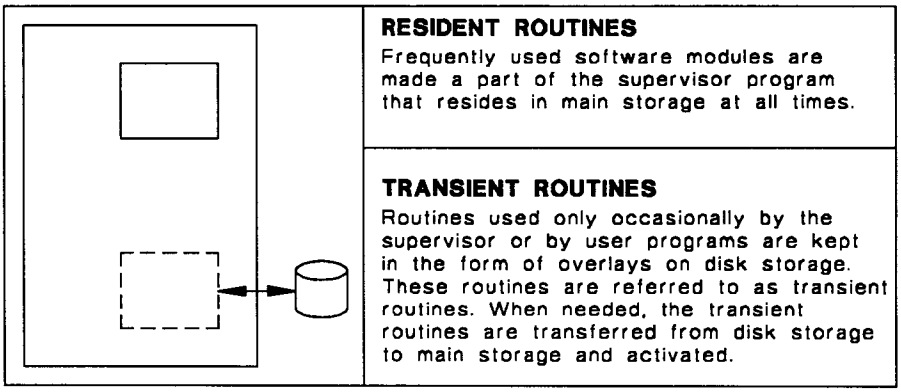
DCF is normally initialized during initial program load (IPL). Commands also let the operator do the following:

- Initialize DCF
- Override defaults
- Activate and deactivate drives to DCF
- Change the cache segment and buffer sizes
- Shut down DCF
- Monitor DCF activity
- Select files to be stored in cache

Transient Management

The supervisor takes full advantage of auxiliary storage to provide maximum services while keeping main storage requirements at a minimum. The supervisor uses two types of routines, resident and transient.

You specify the number of areas in main storage that are set aside to contain transient routines. The supervisor manages these transient areas by monitoring their use, controlling loading of a requested transient routine into a selected area, and transferring control to the transient routine. Only one disk access is required to load a transient routine. Opening and closing files and terminating jobs are examples of transient routines. You can make some frequently used transient routines resident. This reduces load time for the transient routines and the number of I/O operations performed by the system.



A18621

Timer Service Management

The central processor complex contains a high-resolution timer that can provide an interrupt after any time period greater than 1 millisecond. The calling task can specify a wait interval in milliseconds or seconds, or can specify a time of day when an interrupt is to occur.

The supervisor uses a simulated day clock to provide the time of day to tasks upon request and to time-stamp messages and job accounting entries.

System Console Management

System console management controls message displays on the system console, with responses and commands coming from the operator. The screen images scroll upward, with new display lines or operator input appearing on the bottom of the screen. Console management routines selectively delete messages from the top of the screen that do not require responses. Console management also supports and controls additional OS/3 screens available at the console.

Workstation Management

Throughout this overview, the term *workstation* logically means any terminal-like device connected to OS/3 and communicating with an interactive OS/3 product. It does not imply a particular hardware type. The term *terminal* as used throughout this overview refers to communication-connected devices using UNISCOPE protocol, such as SVT and UTS terminals. Personal computers require STEP or some other UNISCOPE emulation. Details on types of devices supported on the System 80 model 7E are supplied earlier in this overview under "Hardware."

Workstation management provides the interface to ICAM for interactive services and jobs that allocate a terminal. It coordinates all traffic to and from the operator.

Error Recovery

When an error causes a program interrupt, the supervisor determines the type of error, such as program check or protection violation, and the type of job involved.

If the supervisor detects an error, it halts the system and attempts a restart from that point. If recovery fails, the supervisor collects information for an orderly abnormal termination.

The supervisor examines machine check interrupts to determine if the error is recoverable. If the error is not recoverable or recovery fails, the supervisor abnormally terminates the system. The supervisor collects and logs all information pertinent to the abnormal termination.

Error Logging

The error logging facility records hardware and software errors in the system error log file. You can retrieve records from this file and use them to prepare statistical reports. The error logging facility is loaded at system generation time. The supervisor then determines the type of errors to be recorded and the devices to be monitored. Error logs are collected for every device configured into the system.

Error log records are collected for four types of errors:

- Peripheral device
- Communications
- Machine check
- User specified

During the initial program load (IPL) of a new session, you can retain the error log file from the previous session or reset the file with the same set of collection parameters or a new set. During operation, you can alter the collection of record types.

Multitasking

Multitasking is a programming technique that significantly reduces the time required to process a job. Each job consists of one or more functions, or tasks, for the system to perform. Normally, each job step has only one task actively using the CPU. Through multitasking, you establish a hierarchy of independent tasks so that several tasks from a single job step are active at the same time. CPU control passes from one task awaiting completion of an external event to another task. If the second task is from the same job, that job gets processed faster. The separate tasks defined for each job step are called subtasks. They vie with tasks from other jobs for CPU time as independent tasks.

Diagnostic Services

Diagnostic services provided by the supervisor include the following:

- Monitor routine – A hardware monitor interrupt lets the monitor routine trace execution of a program to locate and correct errors. The routine interrupts each instruction before it executes and tests for conditions specified in the monitor input. When a specified condition is satisfied (that is, a specified storage location, instruction, or instruction sequence is reached), the routine can print out current program information, a program status word, register contents, the next instruction to execute, etc. The routine can suspend program execution or continue with or without monitor intervention. Trace conditions and information to print are specified through the job control stream or entered at the system console. You can monitor an entire program or a portion of it.

- Snapshot display of main storage – A partial main storage printout aids in solving main storage problems. The area of main storage displayed and the time the display is to occur are identified by parameters at run time or specified in the body of the program.

- Main storage dumps – A main storage dump can provide diagnostics under the following conditions:

  - Abnormal termination dump for user program – This routine provides a main storage dump of a program region in hexadecimal, alphanumeric, or both; plus a formatted display of error codes, job-oriented tables, and supervisor information to assist the user in debugging.

  - Program or operator request dump – This routine lets the operator or any program request a main storage dump in the same format as the abnormal termination dump.

  - System failure dump – This routine is used when an abnormal condition occurs and other dump programs cannot be used.

- Standard system error message interface – An error message service routine provides complete and specific error messages. This routine locates the message in a disk file and transfers control to the system console handler for message display.

- Error response to user jobs – Error codes returned by the supervisor to the calling program are standardized to provide a uniform interface for all system services. If you require the return of control after detection of a hardware failure or software exception, a user-supplied subroutine is provided; otherwise, an orderly abnormal termination, which can optionally include a main storage dump, is called for the user job.

Main Storage Management

The supervisor ensures efficient operation and proper use of main storage by streamlining main storage operations and providing as much automatic storage management as possible. Primarily, the supervisor reserves the main storage space required by jobs, programs, and system routines, and loads them into their reserved areas. The supervisor makes certain that they are loaded into the proper areas and at the proper times. In addition, the supervisor can:

- Dynamically expand the main storage regions of certain programs, jobs, and routines, when necessary
- Rearrange and consolidate job regions to provide as much contiguous free space as possible
- Temporarily suspend and remove jobs from main storage to make room for preemptive priority jobs

Security Maintenance Utility

The security maintenance utility (SMU) controls access to the system's interactive facilities through security information located in the system security file. The system administrator determines who is able to access the system and assigns identification to each system user. The security maintenance utility also accounts for computer time used by interactive system users and automatically executes predefined sets of interactive commands when users log on.

The security maintenance utility performs its tasks by creating three types of profiles:

- A user profile contains security and accounting information, as well as execution and command profile names.
- An execution profile contains the commands for automatic execution at logon, as well as the execution profile name.
- A command profile lets the system administrator specify the system commands a user can enter from a terminal.

The system administrator decides what to include in each of these profiles and can display or modify them.

In addition, the security maintenance utility may restrict access to the system by using a logon procedure that:

- Requires the user to enter a password
- Requires a user profile definition for each individual
- Can allow access only during a specific time or date range
- Prevents a user from logging on after a specified date

Installation Verification Programs

Unisys supplies a set of programs that should be run after SYSGEN is completed to test the functional capability and operation of the various components included on the resident disk unit. These installation verification programs (IVPs) verify that the software you ordered has been installed correctly on your system.

Automated System Power-On and Supervisor Loading

The model 7E system contains hardware and software capabilities that let you automate the system power-on and supervisor loading functions normally done by the operator. You setup and activate these functions via screens and menus displayed at the system console.

Automatic power-on (APO) lets you schedule a system power-on time for each day of the week. You can vary the power-on time and activate or deactivate it on a daily basis. Once activated, the system automatically powers on at the specified time each day without operator intervention.

The automatic boot procedure (ABP) lets you define a supervisor load profile that automatically loads a specified supervisor at the end of system initialization or after a system reset without operator intervention. When ABP is in effect, the system is automatically booted with a default supervisor if the operator does not interrupt the supervisor auto-loading process.

When you use APO and ABP together, you can configure the system to automatically:

- Power on the processor and associated peripherals at a specific time
- Boot the default supervisor
- Execute the console execution profile

You use interactive services to specify the default supervisor, set the timeout interval, and define the console execution profile.

Job Control

Job control manages the system resources and prepares jobs submitted for execution. A job consists of one or more job steps, each requesting the execution of a system or user program.

OS/3 performs job control services at three different times:

- Before execution of the initial job step
- During the transition between job steps
- At the conclusion of the job

You direct job control services through job control language (JCL) statements. These statements define the system resources required for proper execution of a job and facilitate efficient management of these resources. OS/3 JCL is a flexible language that lets you specify the requirements for a variety of essential resources and affords a high degree of independence from limitations imposed by system configurations. Through use of cataloged procedures, OS/3 effectively reduces the usual effort required when running frequently executed jobs.

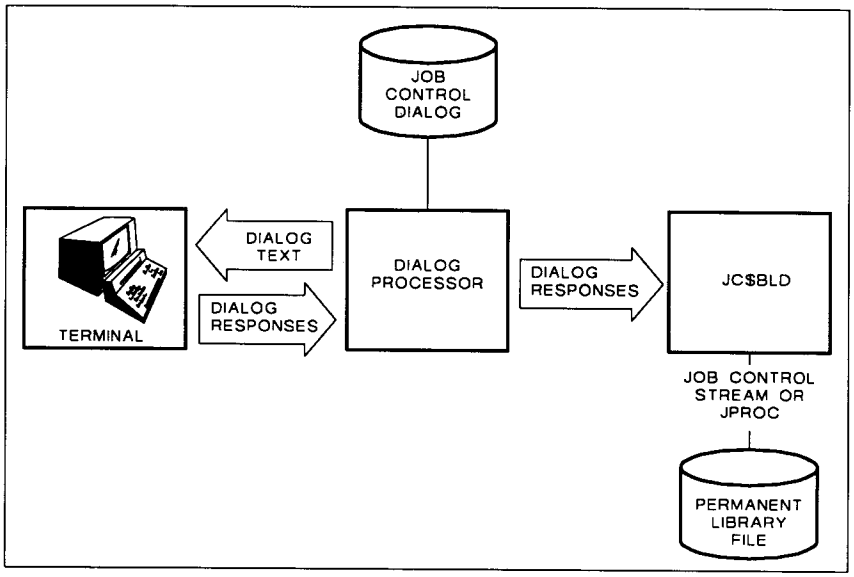
Job Control Services Capabilities List

Automatic job scheduling and initiation
Automatic main storage allocation
Device assignment
Initiation of interactive services for user input
Volume and file label processing
Retrieval of cataloged control streams for subsequent modification and execution
File cataloging
Program restart from a checkpoint
Query of terminal for run-time variables
Scheduling of additional jobs from within a job
Control stream and data storage

Dialog for Job Control Stream and Jproc Preparation

The job control dialog, supplied by Unisys, is a product that leads you step by step through building a job control stream or a jproc from a terminal. It lets you choose the statements and system jprocs required from lists of menu items, and requests parameter values when necessary. An experienced programmer can use this dialog to build a syntactically correct control stream quickly. A novice can use the HELP screens to learn about job control while building a valid job control stream.

The job control dialog automatically stores the control streams and jprocs in either the system job control library file or an alternate library file that you specify.



File Cataloging

The file cataloging facility lets you control file usage and restrict use of selected files to authorized individuals by assigning protective read and write passwords. File cataloging also builds and maintains system generation files. It is a convenient facility because it reduces the amount of job control required to access a file.

The file cataloging facility also lets you alter a file while retaining a copy of it as it was before alteration. This process creates generation files. For example, three generations of a payroll file could be kept: present payroll, week-old payroll, and 2-week-old payroll.

A catalog manipulation utility is provided for the system administrator to perform certain maintenance functions on the catalog.

Interactive Processors

Three interactive processors support the terminal-to-job command interface:

- Command processors
- Dialog processor (except ICAM)
- Screen format coordinator

Command Processors

The command processors provide the interface between the terminal and the various components of the OS/3 software. The command processors handle:

- Commands requesting control system functions, such as logging on and job execution
- Commands issued to control the processing of jobs
- Commands issued to the various interactive facilities

Dialog Processor

Dialogs are used to simplify the process of entering variable data from a terminal directly to a program. The dialog processor:

- Coordinates the display of prefiled dialogs
- Extracts the data entered in response to the dialog
- Routes the data to the appropriate user program for processing

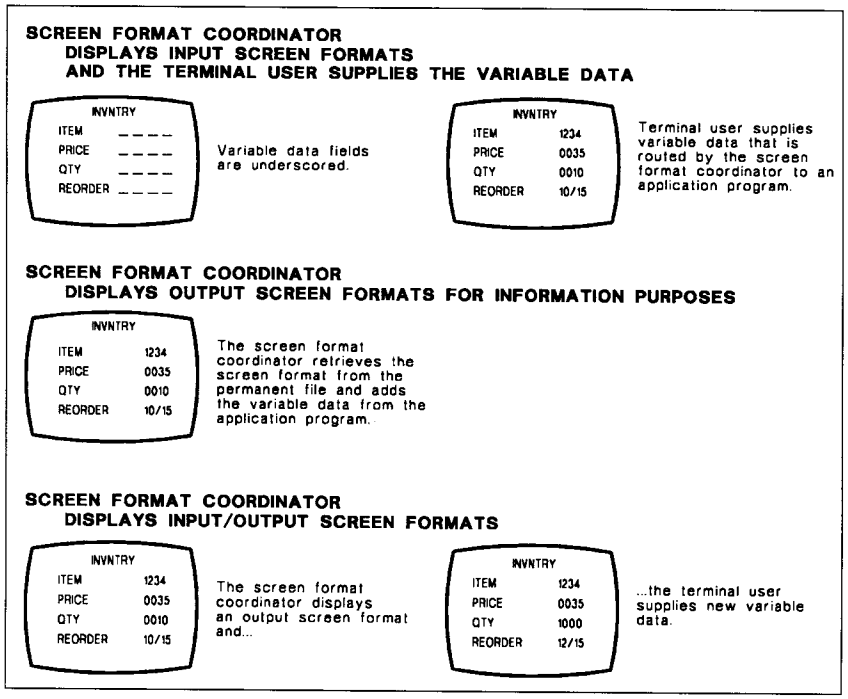
When a program contains Dialog Specification Language (DSL) commands, the DSL translator prompts the dialog processing services to locate and display the specified dialog. The processor maintains an audit file to store entries made to the dialog for future retrieval. You can add to or change the dialog information. The dialog processor produces a printed summary and an audit file of each dialog session. You can use these as guides to change responses in subsequent sessions.

The application programs that solicit input from a dialog session are written in all OS/3 programming languages.

Screen Format Coordinator

Screen formats are forms displayed on the CRT screen. You design the formats, and a program uses them to input data from or output data to a terminal user. Logically, a screen format is made up of rules used to display and verify data the operator has entered. You can define numerous screen and field attributes to facilitate end-user requirements. You can request a screen format through a terminal command, an Information Management System (IMS) action program, an application program, or UNIQUE commands.

The screen format coordinator is activated in response to a program request. It retrieves the appropriate screen formats from the permanent file and displays them with either blank input fields or fields filled in with variable data. The variable data is stored with the program. If a program calls a screen format that serves as both an input and output screen format, the screen format coordinator handles the display of the screen format and variable data, accepts new variable data from the terminal user, and routes that data back to the application program that called the screen format coordinator.



A18823

Consolidated Data Management (CDM)

The consolidated data management (CDM) system is an interface between user programs and the I/O facilities of the supervisor. CDM helps you access data files on peripheral devices without coding routines for blocking/deblocking, buffering, and communicating with PIOCS. The following features are offered:

- A single-access method for disk files
- Shared data management modules
- Device independence
- Support of the interactive features and terminal data transfers

Logical Input/Output Control System

The logical input/output control system (IOCS) modules that control each access method are sharable subroutines. When referenced within a user program, these subroutines are dynamically loaded into main storage and available to many user programs.

Disk Access Method

The CDM access method for disk files is the multiple indexed random access method (MIRAM). MIRAM simplifies data management requirements for disk files by offering a single-access method that provides several ways to access a disk file. The records of a MIRAM file can be accessed as follows:

- Sequentially in order of placement
- Sequentially by ascending key
- Randomly by multiple keys
- Randomly by relative record number

MIRAM conserves main storage space by providing a single data management module for disk access shared by every user program, rather than having several data management modules for each access method. A fast-loader utility allows CDM users to efficiently load keyed MIRAM files. CDM supports access of remote data (but not library) disk files. The processors must be running under the same operating system.

Workstation Access Method

The workstation access method provides the logical level interface between the terminal, the system, and user software. The workstation access method ensures:

- Automatic support for the terminal as an interactive programming device, including support of the interactive features that cause screen displays and management of the screen displays
- Transfer of application data and messages between the terminal and user job
- Console-like capability between the user program, system console operator, and terminal operators
- Support of terminal function key capabilities
- Device independence

The workstation access method is enabled automatically. You can also control it through a set of declarative and imperative macroinstructions that connect terminals at a logical level through the common data interface.

Magnetic Tape and Unit Record Device Access Methods

I/O operations are performed in sequential order on magnetic tape subsystems and printers. Records are handled from first to last, according to physical placement. CDM provides sequential access method modules to handle the access requests of user programs for these device types.

Integrated Communications Access Method (ICAM)

ICAM is designed for the beginning as well as the advanced real-time communications user. A wide range of system options provides support for users committed to a real-time multijobbing environment without penalizing those whose needs are more modest.

ICAM supports the following types of communication:

- Narrow-band transmissions – permit voice-grade communication (dial-up), switched direct distance dialing (DDD), and privately leased lines.
- Wide-band transmissions – permit data transmission over privately leased lines.
- Full-duplex and half-duplex interfaces for Unisys terminals and commercially available data sets

An ICAM configuration includes:

- SYSGEN software modules
- Communications lines
- Terminal devices
- System utilities that the ICAM network requires
- Programs that interface with the network

ICAM features include the following:

- Message queuing
- Multiple destination routing
- Activity scheduling and priority control
- Timer service
- Checkpoint/restart
- Journal control and statistics accumulation
- Trace facility
- Dynamic buffer expansion
- Autodialing
- Public data network support (packet-switched only)

ICAM Components

- Channel control routine – provides the physical I/O interface to data communications data link processors (COMM DLPs) and the specific type of communications subsystems.

- Remote device handlers – provide the software logic and control required to interface the unique characteristics of specific remote devices to the other ICAM components. Some of the remote device handlers available include:

  - Universal Terminal System (UTS 20/30/40/400)
  - SVT 1120/1123/1124 terminals
  - Unisys PCs using UNISCOPE STEP emulation
  - UNIX UNISCOPE emulator
  - Teletype® teletypewriter models 33, 35, 37, and 38
  - BTOS terminals using the UNISCOPE emulator
  - U200/400 and UTS4000 cluster controllers

  Bisynchronous communications procedures are also supported.

- Communications network controller (CNC) – coordinates message flow between the remote device handlers and main storage or a disk-based message queue. CNC places incoming messages on the appropriate processing queues or submits them to special system functions for disposal. CNC also detects a message on an outgoing (destination) queue and provides for its orderly transmission.

  CNC performs the following additional functions, directly or indirectly:

  - Dynamic modification of a network in a changing operating environment
  - Monitoring orderly shutdown of ICAM during end-of-job processing (includes user programs)
  - Scheduling the message user service transcriber routines when new activity is detected and there is an outstanding request waiting

- Distributed communications architecture termination system (DCA TS) – coordinates the flow of messages between local and remote message queues. DCA TS places incoming messages in appropriate message queues or submits them to special system functions for processing. It manages outgoing message queues. DCA TS also protects messages from being lost when communications links break down and provides data throttling and recovery services.

  ICAM DCA TS supports:

  - UTS 20/30/40 terminals
  - SVT 1120/1123/1124 terminals
  - Unisys PCs using UNISCOPE STEP emulator
  - UNIX UNISCOPE emulator
  - BTOS terminals using the UNISCOPE emulator
  - U200/400 and UTS4000 cluster controllers

  ICAM also supports the DCP/5, DCP/15, DCP/25, DCP/30, DCP/35, DCP/40, DCP/50, and DCP/55 running Telcon as remote DCA termination systems, allowing you to function in several modes. The DCP is connected to the System 80 by a universal data link control (UDLC) communications line.

- Communications control area (CCA) – contains the tables required to define and control a specific network configuration. CCAs can be tailored to specific needs.

  ICAM controls a network by setting indicators and flags in the line, terminal, and queue tables within the CCA. These flags control polling, indicate the operational status of communications hardware, and reflect the current disposition of message queues.

  Each CCA contains a pool of network buffers under control of ICAM. Incoming and outgoing messages are temporarily staged in these network buffers during their active transition through the system. Buffers from this pool provide the base when main storage queuing is specified. When disk storage queuing is specified, these network buffers provide intermediate storage during the active input or transmit phase of a message.

- Message user service task (MUST) – provides a message staging service that isolates a message processing program from device dependence. MUST copies input message data from the network buffer pool into designated work areas. It also copies output message data into the network buffers.

  Variations of a MUST routine provide support for specific user program interfaces, such as remote batch processor (RBP), RPG II, IMS, and TIP/30. The MUST routine isolates these functions from the CNC and device handlers, obtaining the maximum commonality of ICAM components.

- Deferred user service task (DUST) – is a series of ICAM overlays performing functions that are neither time dependent nor frequently used. DUST performs the following functions:

  - ICAM initialization
  - Line connect and autodialing
  - Network initialization
  - Program termination
  - Subsystem parameter loading
  - System console message processing

- Global user service task (GUST) – allows formation of a global, or nondedicated, communications network that permits several user programs to be accessed concurrently. The GUST routine controls network and line requests and line releases in a global environment.

ICAM Interfaces

In addition to providing a communications interface between a terminal and interactive services, ICAM also supports the following interfaces:

- Standard message control program interface (STDMCP)
- Transaction control interface (TCI)
- RPG II telecommunications
- COBOL message control system (CMCS)
- Remote batch processing (RBP)
- Direct data interface (DDI)

System Service Programs

OS/3 provides a number of system service programs, including:

- System librarians
- Linkage editor
- Disk and tape initialization routines
- Dump routines
- Disk dump/restore
- Software installation facilities
- System generation facilities

System Librarians

System and user program libraries generate System Access Technique (SAT) files or MIRAM files. OS/3 includes two program librarians, the SAT librarian and the MIRAM librarian, to help you maintain program libraries.

The OS/3 SAT librarian is a set of integrated subroutines that maintains the system and user libraries generated as SAT files. Programs and elements, such as language processor source modules, language processor output (relocatable object) modules, and system-executable load modules, reside within a library, which is either a system or private file in the OS/3 environment.

The SAT librarian performs the following functions:

- Maintenance – The primary task of the librarian is to perform maintenance functions on library modules. These modules consist of entire program libraries, groups of program elements within a specified library, or individual program elements. The librarian can modify existing libraries, create new libraries, and duplicate or purge libraries in their entirety.
- Transfer – The librarian also transfers program library modules from one medium to another. While library modules reside primarily within disk files, the librarian transposes program libraries to or from tape media.

Control statements that perform operations and correct/verify SAT library modules are supplied to the librarian through a control stream. You can print the contents of an individual program module or groups of program modules, or get a table-of-contents-type listing of all the program modules in a file. You can correct or update source modules, as well as sequence them. When the librarian processes a program module, its format is verified.

The MIRAM librarian performs maintenance functions on program libraries generated as MIRAM files. System libraries generated as MIRAM files include the system screen format library and the saved, expanded run library. The MIRAM librarian:

- Copies/deletes all or selected modules from one library
- Prints all or selected modules in a library
- Prints a library directory consisting of all active header records
- Inserts comments into a module header record
- Changes the name or type of a module

The MIRAM librarian responds to a set of user control statements inserted into the control stream that executed the librarian.

Both librarians produce a printed listing of all operations performed during a given librarian session. This listing is called a librarian map.

Linkage Editor

Compilers produce object modules as output. You can tailor these modules into an executable program using the OS/3 linkage editor. The linkage editor can structure object modules from various language processor compilations into a single loadable program. Additionally, the object modules can contain cross-references to each other for the specific purposes of program execution and communication. The linkage editor resolves such cross-references between separately compiled modules when it collects these elements and constructs a loadable program.

Disk and Tape Media Initialization Routines

The various magnetic media initialization, or prep, routines check the condition of the magnetic storage media and prepare them for use by the system. The disk prep routine is used to control alternate track assignments, defect skipping, and surface analysis functions. The tape prep routine is used to control tape density and format (ASCII or EBCDIC), in addition to creating a standard label set.

Dump Routines

The system has a variety of dump routines to aid in the diagnosis of subtle hardware and software errors. These dump routines include the following:

- JOBDUMP – It is possible for a job to terminate abnormally for no apparent reason. The only way to determine the cause of termination is to list and review all information in the system relating to the job. This is called a job dump. Two routines can produce a job dump: the job dump routine and the user end-of-job (EOJ) dump routine. Both routines produce the same information, but the formats are different.

- SYSDUMP – This dump displays the entire contents of storage in hexadecimal format with English explanations of many of the system's tables and structures. The system dump listing is divided into several parts, each part corresponding to a system component.

- SYSDUMPO – This dump performs the same function as SYSDUMP, but allows you to specify what portions of storage are displayed.

- MINIDUMP – This is a dump of only the actual main storage regions involved in the system error or HPR. This is an option of SYSDUMPO.

Disk Dump/Restore

The disk dump/restore utility lets you make backup copies of the contents of a disk volume or volumes. It executes as an interactive program or runs as a batch job. For interactive execution, you enter control specifications for the program in response to a dialog displayed on the terminal screen. Once you provide all appropriate information, the program executes and performs the specified operation.

If the disk dump/restore utility executes as a batch job, through job control, you must prepare a job control stream that includes the appropriate program control statements.

Software Installation Facilities

Unisys delivers the OS/3 SPS already installed on an integrated disk drive and on a set of release tapes. For new systems, there is no installation requirement for SPS.

OS/3 software for program products is delivered on diskettes. Installation of these products prepares your system for support of specific applications and environments. Descriptions of OS/3 program products are supplied later in this overview.

Software installation involves the transfer of program products software to a system disk. Once the software is copied onto the disk, the disk is referred to as the system-resident volume, or SYSRES. The SYSRES disk holds all system software and must remain online when the system is in operation.

Unisys provides installation routines as part of the standard OS/3 release. These routines help you install the following:

- Initial release of OS/3 software
- Any new software received between releases
- Updated software as new releases of OS/3 become available
- System maintenance packages and changes

System Generation Facilities

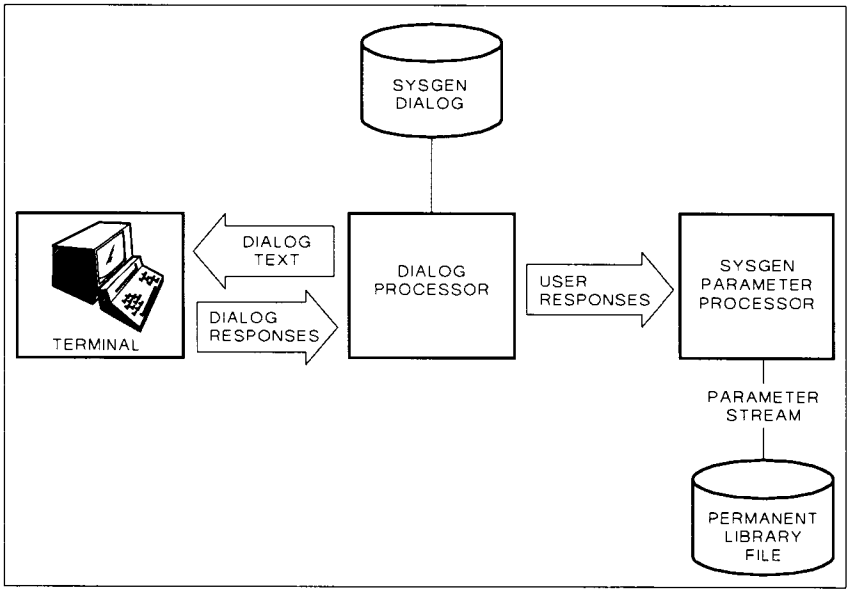
System generation, or SYSGEN, is the process of defining the system's hardware configuration to OS/3 and generating, or creating, the control elements required to meet particular processing requirements. To simplify the procedure, Unisys provides the following SYSGEN facilities:

- Dialog
- Parameter processor
- Job control streams

The SYSGEN dialog is an easy-to-use facility for preparing and processing required parameters. You select SYSGEN parameters in response to queries displayed by the system. OS/3 provides default values for omitted or incorrectly specified parameters to ensure that a usable system is generated.

The dialog accepts your choices and passes them to the SYSGEN parameter processor, which then validates your selections and generates a series of job control streams.

You then execute the actual system generation by interactively running the generated job control streams through simple keyins.



Menu Services

The menu services software supports menus to aid you in data processing activities. Menus are screen displays that list, by number, the options available to you, such as system programs, interactive commands, and other system activities. OS/3 menu services software consists of two components:

- Menu Generator – lets you create, modify, and display menus. You engage in a dialog to create the menu and the HELP screens for all or some of the items offered by the menu. The menus you create can execute programs and interactive services commands, call other menus, and return constant data to a program.

- Menu Processor – retrieves and displays menus on terminal screens, handles user input from the menu, and arranges for program or command execution. It also keeps track of multiple menu sequences. The processor displays the menus in the order specified and responds to user commands to display other menus. Menu processing is available to user jobs through JCL or interactive services.

A Unisys system menu is shown below. Menus list easily executed system programs and interactive services commands. You enter the appropriate selection number to execute the corresponding option.

```
                SYSTEM FUNCTION MENU

1.  DATA UTILITIES                  6.  EDITOR
2.  BASIC                           7.  JCL DIALOG
3.  SCREEN FORMAT GENERATOR         8.  SYSGEN DIALOG
4.  RPG EDIT                        9.  RETURN TO SYSTEM MENU
5.  DDP

                        ENTER SELECTION NUMBER:___
```

Menus also have online HELP screens that provide supplemental information regarding a menu option. The illustration below shows a HELP screen explaining the programming language BASIC, one of the choices on the menu shown above.

```
BASIC (THE BEGINNER'S ALL PURPOSE SYMBOLIC INSTRUCTION CODE)
COMPILER PROVIDES INSTRUCTIONAL SUPPORT AND PROBLEM-SOLVING
CAPABILITIES WITH AN EASILY LEARNED LANGUAGE. BASIC PROVIDES
THE CAPABILITIES OF GENERATING, MODIFYING, SAVING, AND EXECUTING
PROGRAMS. BASIC SUPPORTS ADVANCED LANGUAGE FEATURES SUCH AS
FILES, SUBPROGRAMS, STRING HANDLING, CHAINING, USER-DEFINED
FUNCTIONS, ETC. THE PROCESS OF PROGRAM ENTRY IS INTERACTIVE: THE
WORKSTATION USER IS NOTIFIED OF ANY SYNTAX ERRORS AS THE
STATEMENTS ARE ENTERED.

                        PRESS TRANSMIT TO CONTINUE
```

Interactive Services

OS/3 offers an array of interactive programming features that improve programmer and operator efficiency and help nonprogramming personnel understand and use the system. With interactive services (IS), you can decentralize system operation, develop and execute interactive programs, and initiate system utilities. Several users can access the system simultaneously.

IS offers the following features and benefits:

- Increased productivity – Faster response time and simpler data entry and retrieval methods result in more efficient system usage.
- Easy access – Every authorized member of an organization can access the system through easy-to-use facilities.
- Security – Protective passwords and user identification and command restriction features provide effective security.
- Simplified programming – Interactive communication with the system control software simplifies the entire programming effort.
- Interactive access – Variable data can be entered into jobs currently running in the system.
- Help for novice programmers – Job control dialogs assist novice programmers in building a valid job control stream.
- Help for operators – Soliciting SYSGEN parameters through a SYSGEN dialog assists operators in system generation. A data utilities dialog similarly prompts operators to perform system utilities.
- Screen formats – You can create fill-in-the-blank screens to input, maintain, and display data.
- Multiple jobs – The System 80 model 7E supports 32 job slots, of which 31 are available for user jobs and 1 for IS.

Using terminals as the primary means of communicating with the system, you can use any of the following interactive services:

- General and language editors
- Screen format services
- Dialog processing services

You can issue a set of interactive commands directly from a terminal to:

- Direct operation of the system or a system component
- Query the system for specific information
- Direct execution of a job
- Create and modify source programs, data files, and job control streams

You can use a terminal as an input/output device dedicated to a job or system component, or as a program development tool using various interactive facilities, such as Editor, TPS, ESCORT™ and BASIC. These interactive facilities interface with:

- Information management system (IMS)
- Data base management system (DMS)
- Integrated communications access method (ICAM)
- Distributed data processing (DDP)

Screen Format Services

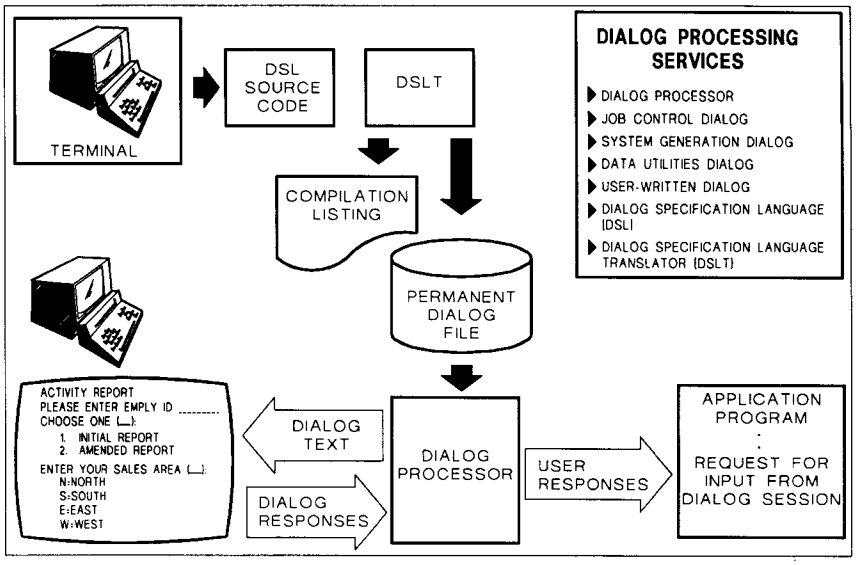
OS/3 screen format services simplifies variable data entry through standardized data input formats. Screen format services lets you:

- Design personalized fill-in-the-blank forms (screen formats)
- Route data entered in response to a screen format back to the application program that requested it
- Display filled-in screen formats to a terminal for informational purposes

Screen format services consists of two software components – the screen format generator, which helps you create screen formats, and the screen format coordinator, which manages screen formats once they are created.

Dialog Processing Services

OS/3 dialog processing services lets programmers and operators communicate easily with programs and procedures by providing interactive, conversational dialogs. Novice operators and nonprogramming users are the primary beneficiaries of the dialog processing services. HELP screens provide tutorial assistance to explain concepts and dialog choices. Responses to the dialogs are routed to the appropriate processor and then stored in a permanent library file.



User-Written Dialogs

You can write interactive dialog text that is displayed later in order to accept varied user responses. This input is routed back to the application program that the dialog complements. For example, a dialog is written to ask the user the type of report desired. The responses cause the program to produce the requested report.

User-written dialogs are written in dialog specification language (DSL). DSL source code is submitted to the dialog specification language translator, which compiles the source code to produce the desired dialog, stores the dialog in a permanent file, and produces a compilation listing.

Transaction Processing

OS/3 provides a new Transaction Platform System (TPS™), which is an integrated online processing and development system. TPS fully supports the development and execution of IMS and TIP/30® action programs and provides the transaction interface to LINC II, OFIS®Link, and CAPIR, the IBM® CICS Application Interface Routine. It contains all the software tools necessary for system maintenance and program development.

For end users, TPS provides a comprehensive, system-wide transaction interface to the OS/3 operating system, allowing all end users to access OS/3 in a consistent and efficient manner regardless of the components utilized.

For applications development programmers, it is designed specifically to improve online system performance and reduce online development time by significantly increasing programmer productivity.

TPS:

- Provides efficient, multithread control of your transaction processing environment.
- Provides a hardware-independent interface to your terminal.
- Provides efficient, secure access to your data files, libraries, and various data base management systems.
- Offers a wide variety of features designed to help end users and program developers, including security, printing, program development and test, and system maintenance.

System Security

TPS provides full system security through user-ids, passwords, assigned security levels, and profile-like applications groups to restrict and control user access to programs and files.

Security is controlled by the Catalog, a file that the system administrator maintains via the online Catalog Manager program. The Catalog contains entries for all authorized users, for all available files, and for all online programs.

Each user has a Catalog entry that specifies the user-id, password, security level, and the application groups that the user can access. Each online program and every online file has an entry that specifies the application group to which it belongs and the security level required to access it.

A user logging on has access only to those facilities available by application group and security level.

Printing Service

TPS provides complete printing services. In addition to sending printer output to the standard OS/3 spool facility for printing, you can redirect data to an online terminal printer, to a Unisys U5000 system, to an OFIS Link document, or to an MS-DOS® file on a personal computer.

PC-to-Host Data Transfer (PCXFER)

Through PCXFER, TPS allows you to design applications that fully integrate PC-based applications and data into your online environment. These applications can be incorporated into existing online and batch systems without the need to redesign the systems.

With PCXFER, the PC becomes one more input source or output destination. Records can be transferred in character or hexadecimal format so that binary data records or even program records can be transferred with no restriction on record length.

A user-developed host file maintenance program can update host files with data read directly from an MS-DOS file. The online file maintenance program has access to the full file journaling and recoder features so that the integrity of the host files is maintained.

Text Editor

TPS offers a powerful and versatile full-screen text editor with a simplified user interface that makes it easy to learn and use. With the full-screen editor, a user can alter displayed text directly or through a complement of online commands.

Integrated Screen Definition, Testing, and Maintenance

TPS includes a facility to define, compose, test, and archive screen formats. You can quickly develop and fine-tune your screen formats to maximize end-user utilization.

Utilities

TPS provides a complete catalog of utility routines to perform a variety of specialized functions. Just a few are:

- CAT – The system administrator uses the online Catalogue Management program to maintain entries in the Catalogue for all users, programs, and files active under TPS.

- CCA – This utility makes ICAM statistical information available to the online user via a series of screen displays or as a printed report. The types of statistics reported include buffer pool penetration, line and terminal traffic, and LOCAP usage for a global network.

- JCL – Users can quickly create and submit one-time-only jobs via the JCL job submission utility. JCL statements are entered at the terminal and submitted directly to the run processor, eliminating the need to create a JCL library element.

- SPL – This utility allows spool file queues to be examined and subfiles to be listed at the terminal. The listing can be manipulated by the user with subfiles printed at the terminal, released to the system output writer, or deleted. In addition, reports created by batch jobs can be easily printed at terminals having printers.

System Activity Monitor (SAM)

The system activity monitor (SAM) measures system efficiency by monitoring and recording various system activities. It is an optional software feature configured during system generation. SAM operates in two modes: monitor class mode and event trace mode.

The monitor measures:

- Central processor and main storage usage
- Disk usage
- Input/output operations
- Supervisor interrupts
- Various communications activities

The information gathered by the monitor allows you to:

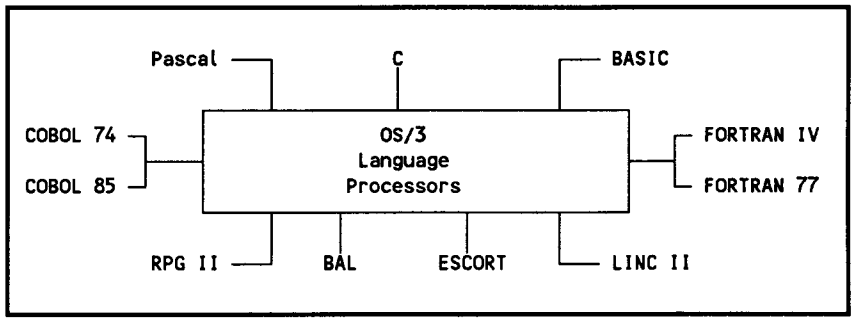
- Detect possible production bottlenecks
- Develop an optimum job mix
- Change system variables to enhance system performance
- Determine the optimum use of the system's peripheral devices
- Make use of capacity planning

Program Products

The System 80 model 7E offers a wide range of program products. Among these products are a variety of programming languages, an interactive general editor, information and data base management facilities, distributed data processing, and interactive system communications.

Languages

OS/3 supports a number of enhanced language processors for use with your System 80 model 7E. These processors provide flexibility in the preparation of a variety of business, scientific, and report oriented application programs.



COBOL 74 and COBOL 85

COBOL is a powerful, general-purpose, English-like language designed for business applications, such as payroll, accounting, inventory, and personnel management.

The COBOL supplied by Unisys also offers a powerful sorting facility that lets users sort data files on multiple keys. In addition, COBOL users can readily interface with IMS and DMS.

The COBOL 74 compiler conforms to the specifications of the *American National Standard COBOL (ANSI X3.23-1974)*. The COBOL 85 compiler is an implementation of the COBOL language as specified in the *American National Standard for Programming Language COBOL (ANSI X3.23-1985, ISO 1979-1985)*.

COBOL 85 is an enhanced and modified version of the previous COBOL standard *(ANSI X3.23-1974)* and has many additional features not available in COBOL 74, notably improved structured programming constructs.

Unisys extensions to the standards for COBOL include floating-point data items, format editing functions, and terminal support.

Pascal

Pascal uses structured programming techniques to improve program development and maintenance. It is designed for professional programmers, students, scientists, and engineers.

The Unisys OS/3 Pascal compiler conforms to the *American National Standard IEEE Standard Pascal (ANSI/IEEE X3.97-1983)*. Because it involves a small number of fundamental programming concepts, Pascal is suitable for teaching programming as a logical and systematic discipline, while providing a practical language for writing systems and application programs.

Unisys OS/3 Pascal also adds several extensions to the standard, including:

- String manipulation
- Relative I/O
- Initial values for variables
- Use of separately compiled functions and procedures
- Conditional compilations
- Multiple exit statements (EXIT, NEXT, RETURN)

C Language

The Unisys OS/3 C compiler is an implementation of the proposed ANSI standard for the C programming language, as described in the "Information Bulletin" published by CBEMA (the April 1985 draft of the X3-J11 Committee Standard).

The C language facilitates the application of structured programming techniques for general data processing problems. Its key features are its small object code size, fast execution speed, and flexibility in building applications.

Unisys extensions include:

- Extensions to the ANSI math library
- Additional signals
- Substitution of macro arguments
- Type conversion options
- Storing in string constants
- Special OS calling sequence

BASIC

BASIC is an interactive programming language that meets the requirements of both business and scientific programming. The BASIC language available on OS/3 complies with the *American National Standard Minimal BASIC (X3.60-1978)* and includes Dartmouth features and compatibility. It has a powerful set of commands that allows the novice to learn the language quickly, yet gives the experienced programmer an extensive list of features for various applications.

BASIC source statements entered directly at the screen are checked for syntax errors as they are entered. A message appears on the screen if a line is in error. Also, BASIC source programs can be compiled directly at the terminal and errors corrected immediately. During an interactive BASIC session, you can input, modify, execute, and save programs.

The OS/3 BASIC compiler performs arithmetic operations, data file processing, matrix generation and processing, and logical operations. You can use subroutines and string operations in a BASIC program.

FORTRAN IV and FORTRAN 77

FORTRAN is a powerful, mathematically oriented programming language geared primarily toward scientific and business applications that require extensive mathematical computing.

FORTRAN language expresses an algorithm in a natural way. You do not need to be concerned with the particular system characteristics. FORTRAN can reference procedures written in another language by name and make them an implicit part of the program.

FORTRAN IV™ is a proper superset of the *American National Standard FORTRAN (X3.10-1966).* It is also a compatible superset of the IBM/DOS 360 FORTRAN IV. FORTRAN 77 conforms to the *American National Standard FORTRAN (X3.9-1978).* These systems feature code optimization, high-performance I/O, and extended functional capabilities.

The primary enhancement of FORTRAN 77 over the previous version is the addition of improved structured programming constructs.

RPG II

RPG II is a high-level language for producing reports and maintaining files in a business environment. The RPG II cycle provides the basic logical flow of a program. You specify the details of processing by using the input format, output format, calculations, and other specifications. You use formatted displays to enter source program statements. Auto report, which accepts simplified RPG II source statements and creates a complete RPG II program, is also available.

To define a report program for generation, you list the requirements on specifications forms. The information coded on these forms becomes the input to the RPG II compiler. The compiler generates an object module for input in the linkage editor. The subsequent output is a loadable program module. You can display the RPG II formats on the terminal screen and complete the proper entries. You can also enter unformatted RPG II source statements.

The following enhancements let you develop more sophisticated report programs within the existing framework of the language:

- Telecommunications interfaces, supplied through an RPG II specifications form, offer extensive communication capabilities to the RPG II user.
- Eight control stream user switches (indicators) set calculations, input files, output files, or specific output records.
- IMS action programs are written in RPG II.
- Terminals are accessible through programmed operations.
- OS/3 RPG II supports single-key and multikey MIRAM files.
- Error diagnostic messages can optionally be written by the RPG II compiler to a module accessed by the editor's error file processor.
- The currency sign (Japanese or other type) can replace the dollar sign in edit words on the control specifications form.

- The RPG II indicator table tests the processing of screen formats. Indicators that are set on when the terminal function keys are pressed are added to the RPG II language.

- Data structures allow multiple definitions of internal data, subdivision of data files, and grouping of fields, with support for packed and zoned decimal data.

- OS/3 RPG II handles multiple terminal and terminal error processing.

- The COPY statement includes RPG statements from a library.

BAL

BAL is a flexible, machine-level language that offers a variety of features and lets you specify the most complex algorithms as a series of mnemonic symbols and command directives. Each mnemonic represents a single machine action. BAL can be of great value to the user with particular programming requirements.

The assembler recognizes a set of directives used to direct operations. These directives control program sectioning, base register assignment, the format of the output listing, sequence checking, and other auxiliary functions.

The assembler includes a macro facility that reduces the effort required to write patterns of coding, either repeated in one program or common among several programs. The macro facility allows a macro to be written so that the pattern of coding generated can vary widely, depending on the parameters supplied with the call. Macro definitions are specified in two formats: MACRO and PROC.

Output from the assembler run consists of a complete listing of symbolic coding, generated object coding, diagnostic messages, and a cross-reference listing. BAL produces a relocatable object module that can link to other modules before it is loaded for subsequent execution.

ESCORT

The ESCORT language is a high-level, interactive programming language that generates file processing, data handling programs, and report generation. It is easy to use, requires minimal training, yet has the power of more complex data handling languages. ESCORT language uses English verbs and clauses to express programming operations. Thus, even a programming novice can quickly begin producing useful programs.

An ESCORT program creates, sorts, or merges files; adds, deletes, sorts, or replaces records within a file; updates selected records from a terminal; issues prompting messages to a terminal for input data; and validates the input data. Available computational operations include addition, subtraction, multiplication, division, and modulo, a division operation in which the result is the remainder only.

You conduct ESCORT programming sessions directly at the terminal. Jobs and programs initiate from the ESCORT call. The ESCORT compiler uses prompting messages for data entry. The amount of prompting depends on the programming expertise of the user.

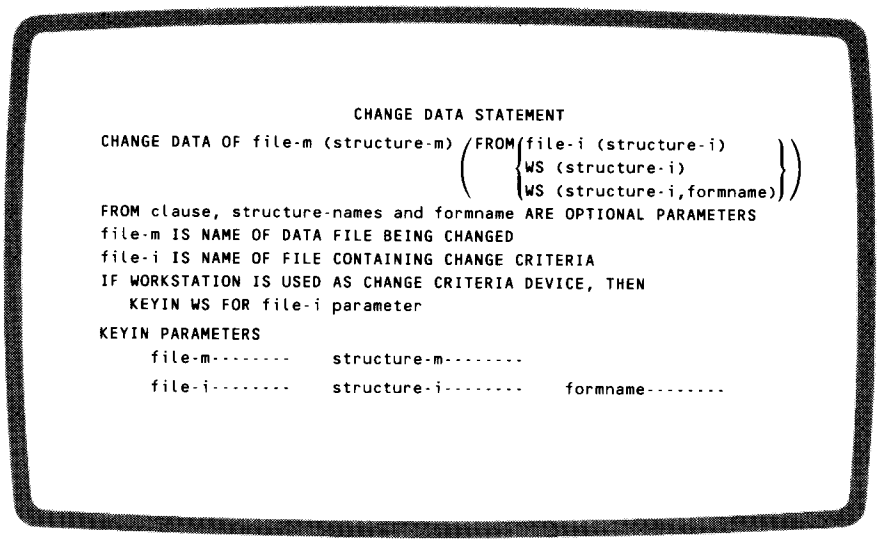
A novice can conduct a session in interactive tutorial mode that creates an ESCORT program using a question-and-answer technique. HELP displays explain various choices offered by the question-and-answer display screens. In tutorial mode, the full ESCORT set of string relational operators is enabled. Each menu has a default option, and you can save programs without exiting the session.

The experienced user can use the interactive program mode to create ESCORT programs faster, with less prompting. Program mode is used to:

- Create programs by filling in the blanks on menu screens displayed by the ESCORT program
- Enter ESCORT statements directly onto a blank screen

Direct entry is the quickest way to create an ESCORT program. The compiler offers immediate syntax checking of ESCORT statements and sends messages when it detects errors. You can correct these errors immediately through the terminal.

Following is a typical ESCORT screen display:



LINC II

Unisys Logic and Information Network Compiler II (LINC II™) is an advanced fourth-generation system generator. With LINC II, you define your information needs in business terms rather than in technical data processing terms.

LINC II uses three basic building blocks:

- The fundamental components of your organization
- The day-to-day business events that occur
- The different ways in which these are viewed by management and user departments (profiles)

Complementing this design philosophy is an implementation language that facilitates the high-level business definition of the information system. A sophisticated interactive development environment assists as the system is defined. LINC II automatically generates every aspect of the system, including the data base, transaction management, network management, and application programs. As a result, analysts can focus on the business problem to be solved rather than on technical implementation details.

LINC II provides the following features:

- Interactive development environment
- Communication between application systems
- Multithreading and synchronized recovery
- Sophisticated reporting capabilities
- Source-level portability across Unisys mainframes
- Efficient resource utilization

General Editor (EDT)

The OS/3 general editor, commonly known as EDT, is a user-oriented interactive program that lets you:

- Create and update library modules, data, and text files from a terminal
- Interactively create and update source programs, job control streams, and data files
- Copy, delete, and concatenate files
- Create and call procedures

It features a comprehensive command set to create and maintain files on disk or diskette. A simple command from the terminal activates EDT.

EDT offers the following versatile features:

- Command language – EDT provides an easy-to-learn interactive command language.
- Procedure files – EDT lets you divide the EDT workspace file into a maximum of 10 subfiles: a main work file and 9 other procedure files. EDT procedure files are basically like the main work file; they consist of data and/or EDT commands. However, unlike the main work file, EDT procedure files can be executed. The commands or data entered in them can be executed against the main work file or any of the other defined procedure files.
- Subeditors – Two subeditors, the RPG editor and the COBOL editor, provide syntax checking on source entry for the RPG and COBOL languages.

Screen commands let you:

- Enter multiple commands and data
- Request help with EDT error messages
- View and update the EDT environment parameters
- Display the EDT commands and their HELP screens
- Return to the EDT session from a screen command
- View and update lines

The editor has file protection facilities that prevent a file from being inadvertently destroyed or altered, either by direct user action or through some system failure.

The general editor operates in two modes: line mode and screen mode. Each mode offers full editorial functions.

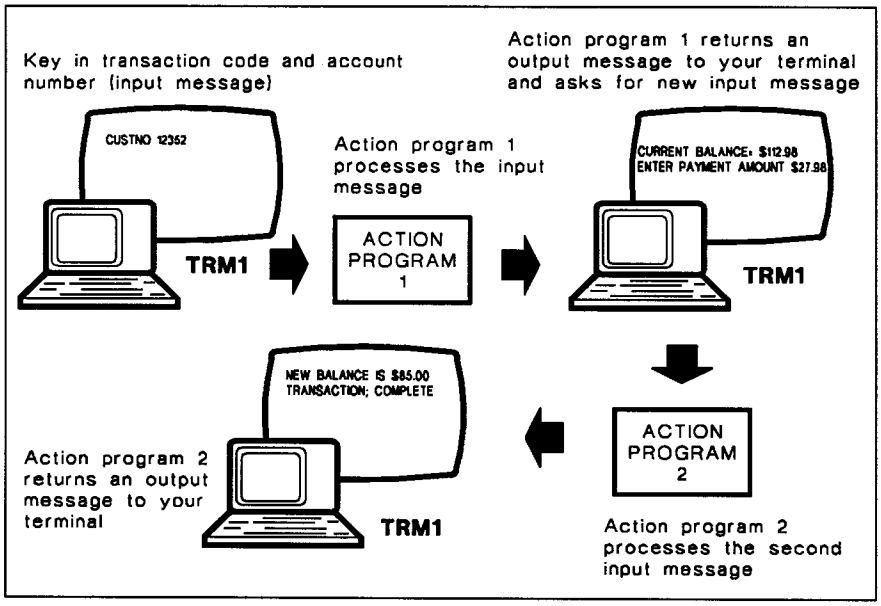
- Line mode – You create or edit material one line at a time.
- Screen mode – You can enter up to 14 lines at a time while using specially formatted screen displays to enter source programs in the COBOL, FORTRAN, and RPG II programming languages.

Information Management System (IMS)

IMS is a transaction-oriented processing system. Each time you enter an input message, such as a transaction code, action programs process the input message, access data files, and provide an output message response to the terminal. IMS operates in a multithread environment where IMS actions from several terminals are processed concurrently. The action program that processes the input message can be written by the user or provided by IMS.

Action Definition

The entire process, from entering an input message to completion of the programming function, is called an action. An action is the basic unit of work in IMS. One or more actions comprise a transaction. When one action accomplishes a task, it is a simple transaction. When you need a sequence of two or more actions to complete a task, it is a dialog transaction. In a dialog transaction, the first action program tells IMS the name of the action program that processes the second message.



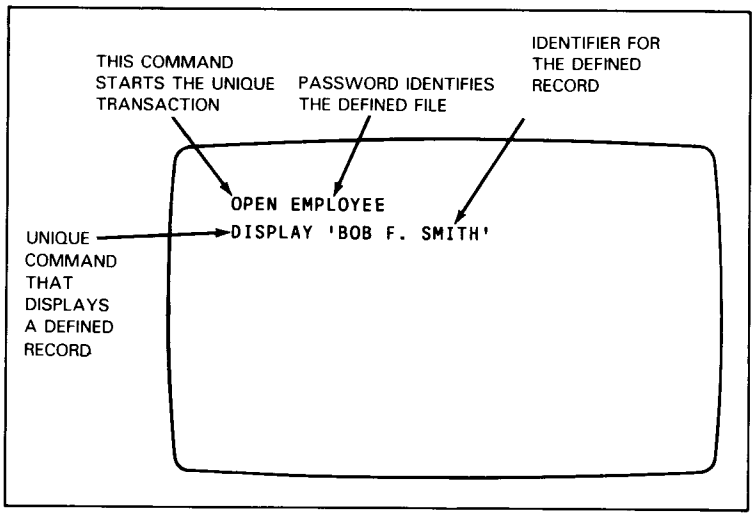
A18627

IMS File Processing Package – UNIQUE

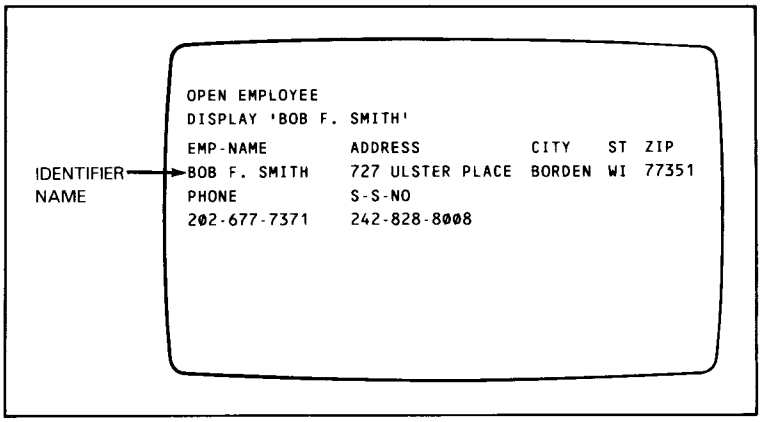
IMS supplies a set of action programs called the uniform inquiry update element (UNIQUE). UNIQUE is a general-purpose language that enables extensive file processing capabilities. Users initiate file activity through UNIQUE commands. In addition to UNIQUE, you can write IMS action programs in BAL, COBOL, or RPG II. This gives you more flexibility in designing IMS programs.

UNIQUE lets you access, retrieve, and update your files. It also lets you calculate statistics on those files. To use UNIQUE, you first create defined files from elements of existing files using an IMS utility called the data definition processor. A password is defined in the data definition that lets you access a defined file. The password is the same as the defined file name, and all configured terminals can use it.

An Input Screen to be Processed by UNIQUE



Output Screen Generated by UNIQUE



Data File Types

IMS can access any type of user disk data file, including:

- MIRAM files – These are used for sequential, direct, or indexed file processing. IMS supports multikey, duplicate key, and key change features to MIRAM files while in consolidated data management (CDM) mode.

- Files generated by DMS – These can also interface with COBOL action programs.

- User-written data definition files – These let you logically redefine a file without actually altering the physical file structure. Redefined files can then be used by IMS action programs.

User-Written Action Programs

IMS also supports user-written action programs in BAL, COBOL, and RPG II. Action programs operate under IMS control and access files through IMS. Programming is simplified because IMS handles all communications and file I/O functions. IMS software allocates resources and schedules action programs to process each transaction.

Both UNIQUE and user-written action programs can access defined files that derive from DMS data base records.

Advanced IMS Capabilities

- IMS/DMS – IMS action programs can concurrently access DMS data bases through data manipulation statements embedded in COBOL action programs.

- IMS/CDM – IMS, in consolidated data management (CDM) mode, lets action programs access records in a DMS data base. IMS loads fixed- and variable-length MIRAM files into a common storage area during system start-up. This eliminates disk reads and writes whenever action programs access records in these files.

- IMS/DDP – In a multithread environment, IMS/DDP gives you flexibility in designing transaction routing to the directory, to action programs, and to the operator.

- Interactive commands - These assist with the administrative and operational aspects of IMS. You can control processing by designating a master terminal to control the processing activity of the entire network through a master terminal command set.

- Language section - You can define a command set for use with UNIQUE. Non-English commands, abbreviated commands, or business-oriented terms are examples of the types of command sets you can define.

- Multikey support - IMS supports multikey, duplicate key, and key change features to MIRAM files during operation in the CDM environment. You can retrieve records from an indexed MIRAM file by using any key of reference. Deletion of records through CDM is also provided for MIRAM files containing single-key or multikey records.

- Statistical reporting - IMS provides statistics, such as the total number of file accesses on a transaction basis, terminal statistics on an IMS session basis, and number of I/O messages per transaction.

- Multiple job environment - IMS can run in a multiple job environment because IMS executes as a program under control of OS/3.

- Dynamic storage allocation - For both single-thread and multithread versions, online IMS provides dynamic main storage allocation at action scheduling time. This lets you modify the size of an action program without reconfiguring it.

- Reliability - IMS provides extensive features to prevent accidental alteration or destruction of data files. Record locking and recovery facilities are also provided.

Data Base Management System (DMS)

DMS accesses and controls data bases stored on direct access devices. There are two main areas of responsibility in DMS:

- Data base administration:
  - Defines data structures (logical and physical) and chooses access methods
  - Monitors system performance
  - Reorganizes the data base as needed to improve system performance
  - Operates DMS utilities to maintain a stable data base
  - Performs start-up/shutdown of the DMS job
- Data base application

  The applications programmer is responsible for writing COBOL/data manipulation language programs using a defined data base structure to interface with the data base.

DMS Components

- Four language processors:
  - Schema processor
  - Subschema processor
  - Device media control language processor (DMCLP)
  - Data manipulation language preprocessor
- The DBMS run-time software component
- System support modules that include utility processors, library files, and procedure files

Data Base Languages

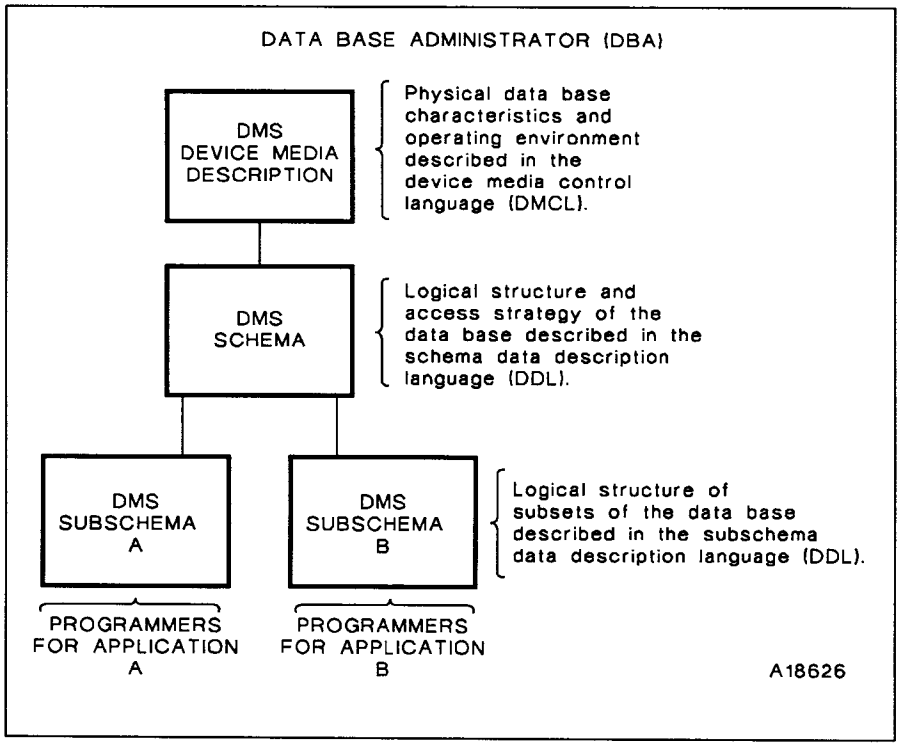
There are two data base languages for data dictionary access:

- Data description language (DDL) – develops a schema to describe the logical structure of the data base, and subschemas to describe data base subsets that relate to specific applications. A DMS data dictionary can contain more than one schema and its related network. DDL is the source input for the schema and subschema processors. The device media control language (DMCL) defines physical characteristics of a user data base and data dictionary. DMCL is the source input for the DMCLP.

- Data manipulation language (DML) – statements are included within conventional COBOL Procedure Division statements for batch processing.

The description of data in a DMS data base is entirely separate from the manipulation of that data in application programs. This results in a higher level of data independence for application programs. All descriptions are written in a high-level language (data description language), which is comparable to the data declaration language of COBOL.

User's View of a DMS Data Base



Run-Time Software

The DMS data base management system (DBMS) is the main set of routines that actually performs the functions requested by a run-unit (either a DMS processor or a user application program). It retrieves and updates data base records conforming to the user-defined data bases and creates a journal file for use in offline manual recovery.

The DBMS is a single reentrant run-time component running as a privileged job under OS/3.

System Support Functions

The system support functions create, establish, and maintain a data base. There are three categories of system support functions:

- Configuration function – creates an installation version of DMS that fulfills specific processing requirements.
- System generation function – creates and initializes a data dictionary and a data base description.
- System maintenance function – is a recovery-related utility processor. The DMS utility processor language lets you specify the utility functions to be performed and the portion of the data base they operate upon. DMS recovery utility functions include journaling, security dumping, automatic and manual recovery methods, and DBA processors that print reports and statistics.

Advanced DMS Capabilities

- You can exclusively lock a record as soon as it is retrieved to prevent record update conflicts.
- The action program can test the availability of a data base record before retrieving it. If the record is not available, a minor end is returned rather than a fatal thread cancellation.
- You can sequentially process the same data base using two or more DMCLs.
- DBPAG prints a compromised data base page instead of terminating with an error message. This allows analysis of hardware or software problems suspected of causing the compromises.
- DMS allows 32 index keys per record.

- You can specify a unique journal file for each data base. This allows concurrent operation of journaling and use of multiple data bases that have different page sizes.
- Disk journal files can extend over more than one DBMS session.
- You can obtain a system dump when an unexpected data management error occurs.
- The DBMS log records the activity of the DBMS.
- DBMON provides a *window* into the DBMS run-time component during execution through 12 monitor screens. Data on the monitor screens allow you to analyze and improve DBMS performance.

Distributed Data Processing (DDP)

OS/3 DDP lets you:

- Distribute files between remote data processing systems
- Engage those systems to cooperatively process jobs
- Access files and other programs on remote systems
- Perform transaction processing between remote systems

DDP functions in OS/3 are divided into three software packages, each offering different elements. You can tailor distributed processing software to include only the particular features your installation requires. The software packages are:

- DDP transfer facility
- DDP file access
- IMS-DDP transaction facility

DDP Transfer Facility

The DDP transfer facility lets you view each system in the DDP network as an available resource for scheduling and executing your work. Using simple commands, you can initiate job distribution and file transfer within the system without concern for the requirements of the hardware and software of each system, and without concern for the communications protocols needed to initiate and monitor the distribution of a job.

The DDP transfer facility provides the following capabilities:

- Site-to-site data file and program library transfers
- Remote job initiation and control
- Operator console control over remote sites and routing of messages to remote consoles
- Routing print output files to different host computer systems
- Routing print output files to remote auxiliary printers

The DDP transfer facility consists of two pieces of software, the job transfer manager and the file transfer manager.

Job Transfer Manager	File Transfer Manager
Provides commands to:	Permits you to:
- Submit jobs to the DDP network	- Generate a file directory to catalog the characteristics of files
- Monitor execution of submitted jobs	- Transfer copies of data files and program libraries
- Cancel a submitted job	- Delete a file from the file directory
- Communicate with a remote operator's console	- Obtain a listing of a file's characteristics from the file directory
- Issue instructions to a remote operating system	
- Respond to messages issued by a job executing on a remote system	

Files Transmitted between Systems

- Data files – Files transmitted from OS/3 are MIRAM files. Files transmitted to OS/3 will be created in MIRAM format.
- Program libraries – Any directly accessible module in a program library or an entire library can be transferred between systems.

DDP File Access

DDP file access lets a user program, through OS/3 JCL, access files residing on remote OS/3 systems within a DDP network. DDP file access also lets user programs exchange data and control information. You can transfer program control between systems and reroute spooled output.

DDP file access provides the following capabilities:

- User programs can access disk files residing on a remote OS/3 host through job control statements and use them as if they resided on the local system.
- Programs running on remote hosts in a multihost environment can access each other as files.

IMS-DDP Transaction Facility

The IMS-DDP transaction facility lets you perform transaction processing on any local or remote files within a DDP network. The transactions are processed by IMS user action programs written in BAL, COBOL, and RPG II, or by the UNIQUE inquiry language.

The IMS-DDP transaction facility also lets you perform inquiries and updates on files residing at remote terminals anywhere within the DDP network.

IMS-DDP software requires a configured communications network to physically link the DDP processors at each site. The processors interpret all DDP-related commands and perform the requested functions. The processors are designed to operate in an interactive environment and respond to commands issued from a terminal.

There are three ways in which IMS can route a transaction to a remote system:

- Directory routing – A terminal operator at the primary IMS enters a transaction code that identifies a transaction at a particular remote system. The primary IMS routes the message to the secondary IMS, where UNIQUE or action programs process the transaction. Once the transaction begins, a communications link is established between the terminal operator and the remote system. This allows a dialog transaction consisting of multiple input and output messages.

- Operator routing – Operator routing is similar to directory routing. In this case, a special character, rather than a transaction code, is associated with the remote system. When the terminal operator enters a transaction code prefixed by the special character, IMS routes the message to the secondary IMS, where the transaction is processed in the same manner as in directory routing.

- Action program routing – The terminal operator enters a transaction code that initiates a transaction at the primary IMS. A COBOL or BAL action program at the primary IMS issues an ACTIVATE function call to IMS, identifies the remote system in its output message header, and generates a message containing a transaction code. IMS routes this message to the remote IMS, where action programs process the transaction and return a message to the originating action program or its successor. The action program at the primary IMS can then return a message to the terminal operator or can issue another ACTIVATE call to initiate another remote transaction.

PC-to-OS/3 Connectivity

The Unisys personal computer (PC) user has an increasing need to access System 80 data bases and software applications for daily business use. Unisys provides three utilities to connect PCs to System 80:

- File transfer (PCTRAN)
- On-line transfer (OLTRAN)
- PC on-line disk

PCTRAN

PCTRAN permits transfer of data files, library modules, and source program files between a disk or diskette on a PC and a disk, diskette, or tape on System 80.

You can select character or PC hex mode data transmission. When using hex mode, the PC terminal emulation software and PCTRAN translate the data during file transfer. This option allows PC object code and binary data to be transferred to and from the OS/3 host system.

When you transfer OS/3 object or load modules to a PC, the data is translated by PCTRAN and maintained on the PC as character data. When the module is retransmitted to the host system, PCTRAN translates it into OS/3 object code format.

OLTRAN

OLTRAN is an OS/3 program product that offers you an easy way to transfer files between a PC and a System 80 mainframe. You can transfer OS/3 MIRAM files and SAT file elements to and from PC ASCII data files. You can transfer ASCII files from a PC to the OS/3 mainframe with or without translation to EBCDIC code. A complete error-checking protocol is provided to ensure error-free data transfer.

PC On-line Disk

PC on-line disk lets you allocate and format OS/3 files as Unisys PC diskettes or hard disks. Multiple PCs can use the same disk files, allowing users to share data, keep backup copies of data diskettes on an OS/3 host, and assign an area of random access memory (RAM) for use as a diskette.

This utility supports most MS-DOS PC operating commands and MS-DOS compatible software.

File Placement Analyzer (FIPLAN)

FIPLAN can improve system performance by projecting optimum file allocation. The projections are based on the contents of the volume table of contents (VTOC) when SAM is initialized. FIPLAN can analyze and project allocations for up to 500 files at a time. The allocation of files recommended by FIPLAN can result in the relocation of user files across volume boundaries to achieve optimum load balancing. FIPLAN does not analyze dynamic file extensions and new files created during the monitor session.

UNISYS

70041785-100

Printed in US America

11/91

7004 1785-100