

PUBLICATIONS REVISION
System 80
Models 10-20 Capabilities Overview
UP-12443 Rev. 3

This Library Memo announces the release and availability of the *System 80 Models 10-20 Capabilities Overview*, UP-12443 Rev. 3.

This overview is a standard library item (SLI). It is part of the standard library provided automatically with the purchase of the product.

This overview describes the hardware and software of the System 80 models 10-20 data processing systems. System 80 models 10-20 feature high-quality, compact, integrated hardware, complemented by advanced, easy-to-use, interactive software. They use Operating System/3 (OS/3).

This revision describes the LINC II and MAPPER C fourth-generation languages, adds COBOL 85 and C to the list of language processors System 80 supports, and provides descriptions of the high-performance 9246-25 nonimpact printer and the 9720 disk subsystem.

Additional copies may be ordered through your Unisys representative.

Destruction Notice: This revision supersedes and replaces *System 80 Models 10/15/20 Capabilities Overview*, UP-12443 Rev. 2, released on Library Memo dated April 1988. Please destroy all copies of UP-12443 Rev. 2 and its Library Memo.

LIBRARY MEMO ONLY	LIBRARY MEMO AND ATTACHMENTS	THIS SHEET IS
Mailing Lists SAB and SAE	Mailing Lists MB00, MB01, and MBW (116 pages plus Memo)	Library Memo for UP-12443 Rev. 3
		RELEASE DATE: May 1989



UNISYS

**System 80
Models 10-20**

**Capabilities
Overview**



OS/3 Release 13.0

May 1989

Priced Item

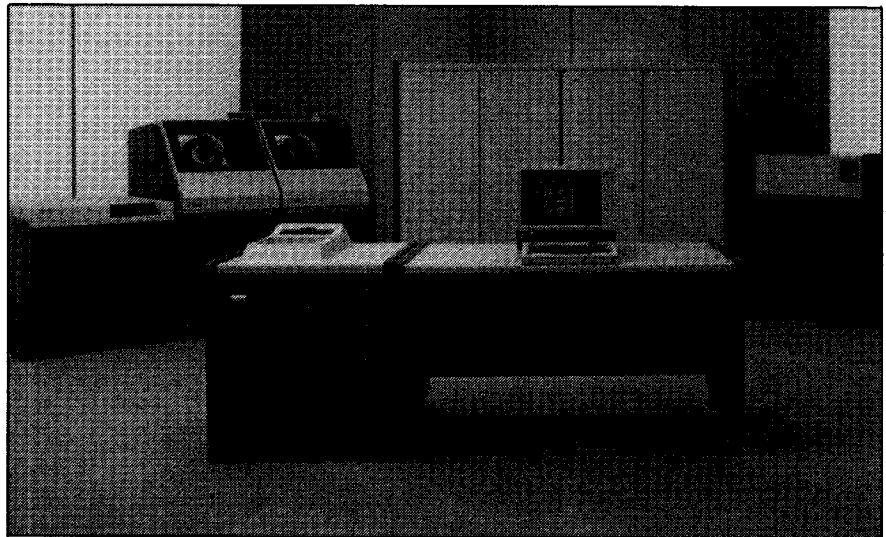
Printed in U S America
UP-12443 Rev. 3



UNISYS

System 80 Models 10-20

Capabilities Overview



Copyright © 1989 Unisys Corporation
All rights reserved.
Unisys is a trademark of Unisys Corporation.

OS/3 Release 13.0

May 1989

Priced Item

Printed in U S America
UP-12443 Rev. 3

NO WARRANTIES OF ANY NATURE ARE EXTENDED BY THE DOCUMENT. Any product and related material disclosed herein are only furnished pursuant and subject to the terms and conditions of a duly executed Program Product License or Agreement to purchase or lease equipment. The only warranties made by Unisys, if any, with respect to the products described in this document are set forth in such License or Agreement. Unisys cannot accept any financial or other responsibility that may be the result of your use of the information in this document or software material, including direct, indirect, special or consequential damages.

You should be very careful to ensure that the use of this information and/or software material complies with the laws, rules, and regulations of the jurisdictions with respect to which it is used.

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

Correspondence regarding this publication should be forwarded directly to Unisys Corporation, to E/MGSG Product Information, P.O. Box 500, M.S. E5-114, Blue Bell, PA 19424 U.S.A.

Contents

Section 1. Introduction

Advanced Hardware Technology	1
Integrated, Interactive Software	2
Peripheral Compatibility	2
System Reliability	2

Section 2. System Software

System Control Software	3
Supervisor	4
Supervisor Interrupt Requests	6
Task Switcher	7
Physical Input/Output Control System (PIOCS)	7
Disk Cache Facility (DCF)	8
Transient Management	9
Timer Service Management	10
System Console Management	10
Workstation Management	10
Error Recovery	10
Error Logging	10
Multitasking	11
Diagnostic Services	11
Main Storage Management	12
Security Maintenance Utility	13
Installation Verification Programs	13
Job Control	14
Dialog for Job Control Stream and Jproc Preparation	15
File Cataloging	16
Consolidated Data Management (CDM)	17
Logical Input/Output Control System	18
Disk Access Method	18
Diskette Access Method	18
Workstation/Terminal Access Method	18
Magnetic Tape and Unit Record Device Access Methods	19
Device Independence	19
System Service Programs	19
System Librarians	20
Linkage Editor	21
Disk, Diskette, and Tape Initialization Routines	22
Dump Routines	22
Disk Dump/Restore	22

Disk Copy	23
Catalog Manipulation Utility	23
Software Installation Facilities	23
System Generation Facilities	23
Extended System Software	25
Data Utilities	25
Spooling and Job Accounting/Reporting	26
SORT/MERGE	27
IBM Compatible Sort	28
Menu Services	29
Interactive Services	30
Workstations	31
General Editor	32
Screen Format Services	33
Screen Format Generator	33
Screen Format Coordinator	34
Dialog Processing Services	35
Enhanced Programming Languages	36
COBOL 74 and COBOL 85	36
Pascal	37
C Language	37
BASIC	38
FORTRAN IV and FORTRAN 77	38
RPG II	39
Basic Assembly Language (BAL)	40
ESCORT	40
Integrated Communications Access Method (ICAM)	42
ICAM Features	42
ICAM Components	43
ICAM Interfaces	45
Nine Thousand Remote (NTR) System Utility	46
Information Management System (IMS)	46
Action Definition	46
IMS File Processing Package - UNIQUE	47
Data File Types	47
User-Written Action Programs	47
IMS Interface Capabilities	47
Advanced IMS Capabilities	48
Data Base Management System (DMS)	49
DMS Components	49
Data Base Languages	50
Run-Time Software	51
System Support Functions	51
Advanced DMS Capabilities	51
Distributed Data Processing (DDP)	52
DDP Transfer Facility	53
Job Transfer Manager	53
File Transfer Manager	53
Files Transmitted between Systems	54
DDP File Access	54
IMS-DDP Transaction Facility	54

Fourth-Generation Languages	55
LINC II	55
MAPPER System	56
MAPPER 5-to-OS/3 Connectivity	57
MAPPER 80	57
PC-to-OS/3 Connectivity	57
PCTRAN	57
OLTRAN	57
PC Online Extract	58
PC Online Disk	58
Maintenance and Diagnostic Software	58
Resident Diagnostics	58
Offline Diagnostics	59
Online Diagnostics	59
System Maintenance Packages	59
System Activity Monitor (SAM)	59
File Placement Analyzer (FIPLAN)	60
UTS 400/4000 Support	60
UTS 400 COBOL	60
UTS 400 Edit Processor	60
UTS 400 Load/Dump Facility	61
Multijob Processing	61
Batch Job Processing	61
Interactive Job Processing	61
Dynamic Resource Management	62
Resource Management Product	62

Section 3. Hardware

Central Processor	63
Basic Processing Unit (BPU)	63
System Control Processor	64
Mainframe Interface Adapter	64
Main Storage Unit	64
Operand and Instruction Caches	64
Floating-Point Processor	64
I/O Facilities	65
I/O Processor	65
Selector Channel	65
Instruction Set	65
Nonprivileged Instruction Set	65
Privileged Instruction Set	65
Hardware Configurations	66
Basic System	67
System Growth	67
Migration from Other Systems	68
Peripheral Subsystems	68
Disk Subsystems	68
8416/8418 Disk Subsystems	69
8430/8433 Disk Subsystems	70
8470/8480 Disk Subsystems	71

8417 Nonremovable Disk Drive	72
8419 Removable Disk Drive	73
8494 Disk Subsystem	74
9720 Disk Subsystem	75
Workstation Subsystems	76
Workstation Capabilities	76
Local Workstations	76
Model 1 Workstation	76
Model 2 Workstation	77
Workstation Printers	78
Remote Workstations	80
System Console	80
Alternate Console	80
Printer Subsystems	81
0789 Line Printer	82
9246-14B High-Performance Line Printer	83
0776 High-Performance Line Printer	84
0770 Line Printer	85
0770 II Line Printer	86
0798 Character Printer	87
Magnetic Tape Subsystems	88
UNISERVO 10 Magnetic Tape Subsystem	88
UNISERVO 10/14 Magnetic Tape Subsystem	89
UNISERVO 12/16 Magnetic Tape Subsystem	90
UNISERVO 20 Magnetic Tape Subsystem	91
UNISERVO 22/24 Magnetic Tape Subsystem	92
UNISERVO 26/28 Magnetic Tape Subsystem	93
3200 Series Magnetic Tape Subsystem	94
Streaming Magnetic Tape Subsystem	95
Diskette Subsystem	96
T8420/8422 Diskette Drives	96
Punched Card Subsystems	97
0716 Card Reader Subsystem	97
0719 Card Reader Subsystem	98
0608 Card Punch/Reader Subsystem	99
Communications Subsystems	100
UTS 20	100
UTS 30	101
UTS 40	102
UTS 400	102
UTS 4000	102
SVT 1120 Terminal	104
SVT 1123 Remote Workstation	104
SVT 1124 Remote Workstation	104
DCP 15/30/40/50	105
Additional Communications Devices	105
Hardware Summary	106

Section 1

Introduction

High-quality, integrated hardware complemented by an advanced yet easy-to-use interactive software package – that's what you get with the Unisys System 80.

A prime consideration when purchasing a computer system is the amount of productive computing time you can expect. To ensure continuing reliability and maintainability of its systems, Unisys has included extensive automatic recovery and maintenance features in the hardware and software design for System 80.

The hardware represents the state of the art in both technology and reliability. Extensive testing before shipment ensures that any installed hardware is in perfect operating condition.

The OS/3 software has proved itself through years of dependable service. It features a full complement of integrated, interactive software products.

As your business grows, you can expand System 80 easily by adding peripheral devices and increasing main storage capacity.

Advanced Hardware Technology

System 80 incorporates a number of advanced hardware features that increase system efficiency and enhance productivity.

- Large-scale integrated circuits increase system speed and processing power.
- A four-stage pipeline instruction processor with a multi-instruction buffer decreases execution time.
- Independent function microprocessors improve throughput processing.
- High-density, high-capacity disk devices offer you expandable mass storage facilities.
- A variety of high-speed printers gives you flexibility in selecting character sets and line lengths to provide maximum printing capacity.
- Reliable, high-density tape drives, including streaming tape drives, make Unisys tape systems valuable mass storage devices for all your needs.

Integrated, Interactive Software

System 80 provides many features that uniquely qualify it as a powerful, flexible, and total business data processing system.

- A full range of interactive services
- Dialog processing services
- A variety of programming languages
- Information and data base facilities
- Multijob processing
- Dynamic resource management
- Integrated data communications
- Distributed data processing
- System maintenance packages

Peripheral Compatibility

If you have Series 90/25, 90/30, 90/40 systems, or System 80 models 3-6 or model 8 systems, you can upgrade to System 80 models 10-20 and connect many of your present peripheral devices, including disk devices, tape drives, card readers, and printers. For further information on which peripherals can be connected to System 80 models 10-20, see "Peripheral Subsystems" in Section 3.

System Reliability

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

- An array of diagnostic facilities to detect any possible software and 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

Section 2

System Software

OS/3 directs the efficient and flexible centralized control of all System 80 activities. It gives you a multijobbing 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 local and remote workstations.

OS/3 is designed to take advantage of the increased speed and capacity of System 80 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 in Section 4.

OS/3 is a comprehensive library of software that supports System 80 at three levels.

- System control software (SCS) – consists of the facilities you need to load, operate, and maintain the minimum System 80 operating environment. SCS is required for operation of any of the other proprietary system software products.
- Extended system software – is built on the basis of SCS. It includes the wide range of Unisys proprietary software programs that extend the usability of System 80. Extended support software includes language translators, utility programs, and general application programs.
- Additional software – supports special-purpose services, such as hardware diagnostics, performance evaluation, and migration/conversion programs.

System Control Software

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

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

Supervisor

The supervisor is a collection of sophisticated routines that provide 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

- Multiprogram 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 System 80 an easy-to-use and efficient data processing system.

The supervisor is built around executable modules, or routines, each of which has a specialized function. Resident routines are commonly used by the supervisor and always reside in main storage. Other less frequently used routines, called transients, are stored on the system-resident volume and are loaded in main storage only when the supervisor needs them. This arrangement minimizes the amount of main storage the supervisor uses by overlaying unneeded transients with newly loaded transients. It also eliminates the I/O time needed to load the most commonly used routines by keeping them resident.

The supervisor is configured during the system generation procedure (SYSGEN). During SYSGEN, you enter parameters to indicate the supervisor routines you want to include as resident or transient. Additional features increase the main storage requirements of the supervisor.

A number of supervisor generation parameters provide default values for handling certain conditions. You can fast-load the supervisor and generate a number of separate supervisors during system generation; but you can only operate one supervisor at a time.

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 workstation management	- Controls information to and from the system console and various workstations configured.
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	- Isolates and corrects diverse error conditions.
Main storage management	- Allocates and deallocates main storage as required by other system control software.
Security	- Controls access to the system.
Installation verification	- Tests functional capability of system components.

Supervisor Interrupt Requests

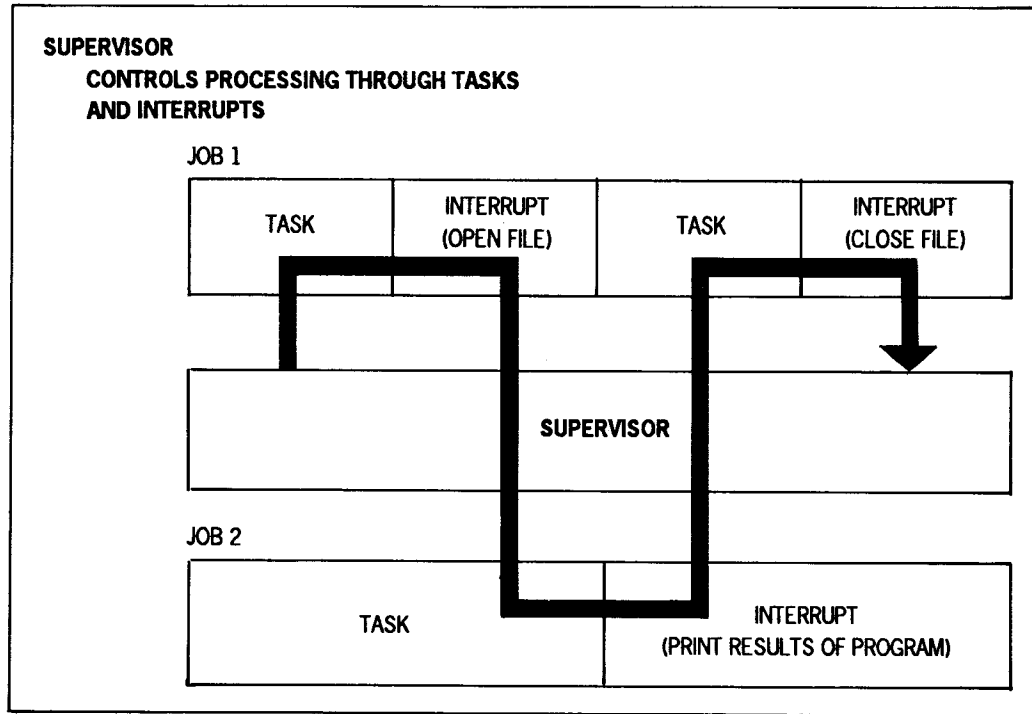
A supervisor interrupt is a request made to the supervisor to perform a function. Requests are generated by the various hardware and software components. Interrupts stop the normal processor flow and must be completed before processing can continue. OS/3 recognizes eight types of interrupts.

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 PIOCS, which is composed of general-purpose software routines that provide maximum throughput on all peripheral devices and allow expansion to support new devices.

PIOCS handles queuing and initiation of all I/O commands, and 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** – This 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, a general I/O status analysis is performed to determine whether an abnormal condition occurred.

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

Disk Cache Facility (DCF)

DCF increases system performance by reducing the I/O bottleneck to disk devices. I/Os are slow relative to processor speed. The I/O bottleneck is reduced by reducing the number of physical I/Os.

When a particular area on a disk is read, DCF 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 I/Os required 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)
- **Process time overall**

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.

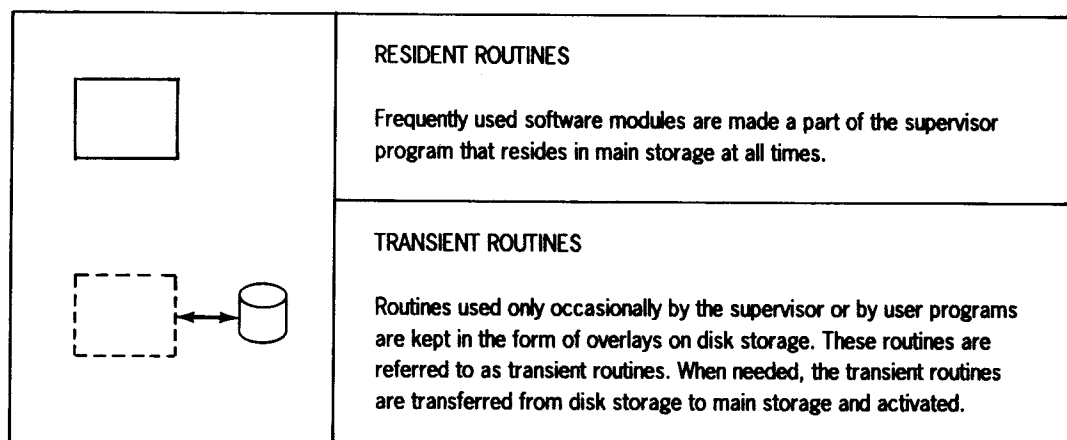
DCF is normally initialized during initial program load (IPL). Commands also let the operator do the following:

- Initialize DCF
- Override defaults
- Activate and deactivate drives to DCF
- Change the cache segment and buffer sizes
- Shutdown 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 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.



Timer Service Management

The central processor complex contains a high-resolution timer that can provide an interrupt after any time period greater than one 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.

Workstation Management

Workstation management provides physical level support for the interactive services and data transfers for workstations dedicated to a job. To ensure efficient operation, workstation management coordinates all the activities of the workstation, or terminals emulating a workstation, with those of the requested functions.

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.

Optional, user-supplied subroutines process error information. Standard error control actions are initiated in the absence of user code. 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 monitor can print out current program information, a program status word, register contents, next instruction to execute, etc.; and 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** – This service routine provides complete and specific error messages. It 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 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 the system's main storage. 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

The supervisor performs main storage management functions to promote system efficiency and ease of use by streamlining main storage operations and providing as much automatic storage management as possible.

Security Maintenance Utility

The security maintenance utility 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 workstation.

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

In addition, the security maintenance utility restricts access to the system by use of the logon procedure which

- 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 complete to test the functional capability and operation of the various components included on the resident disk unit.

A verification routine performs a thorough analysis of a volume table of contents (VTOC), looking for any inconsistencies between the labels. When the routine detects an inconsistency, it displays a message providing the details on the error. When no inconsistencies are found, the routine informs the user that the volume is free of problems. VTOC verification can be performed at automatic volume recognition time or through a console command.

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

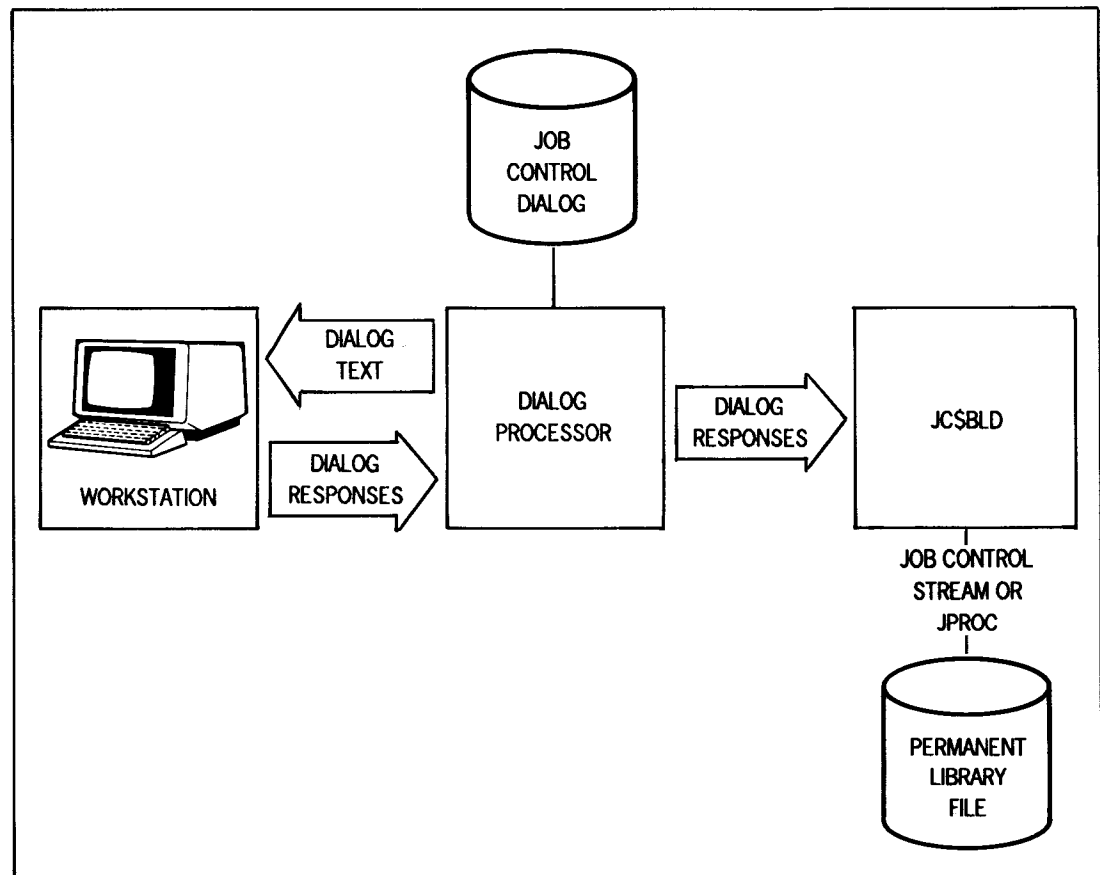
A control stream is a group of sequenced statements, written in JCL, that defines a job and directs its execution. The statements are divided into four functional groups.

- Job coordination
- Device assignment
- Job step operation
- Regulation of job environment

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 workstation. It lets you choose the statements and system jprocs required from lists of menu items and requests parameter values where 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 (\$Y\$JCS) or an alternate library file that you specify.



File Cataloging

The file cataloging facility provides a method to control file usage and to restrict use of selected files to only authorized individuals. File cataloging also builds and maintains generation files. It is a convenience facility because it reduces the job control required to access a file.

To use the cataloging facility, the device assignment sets of the cataloged files are placed in the system catalog file. A device assignment set consists of the job control statements that define a file and its location. To place a device assignment set into the catalog, include a cataloging statement in the device assignment set.

If use of a cataloged file is restricted, protective read and write passwords are assigned to the file. A read password assigned to a file restricts access to the file. A write password assigned to a file restricts changes to the file by unauthorized users.

The file cataloging facility 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 two-week-old payroll.

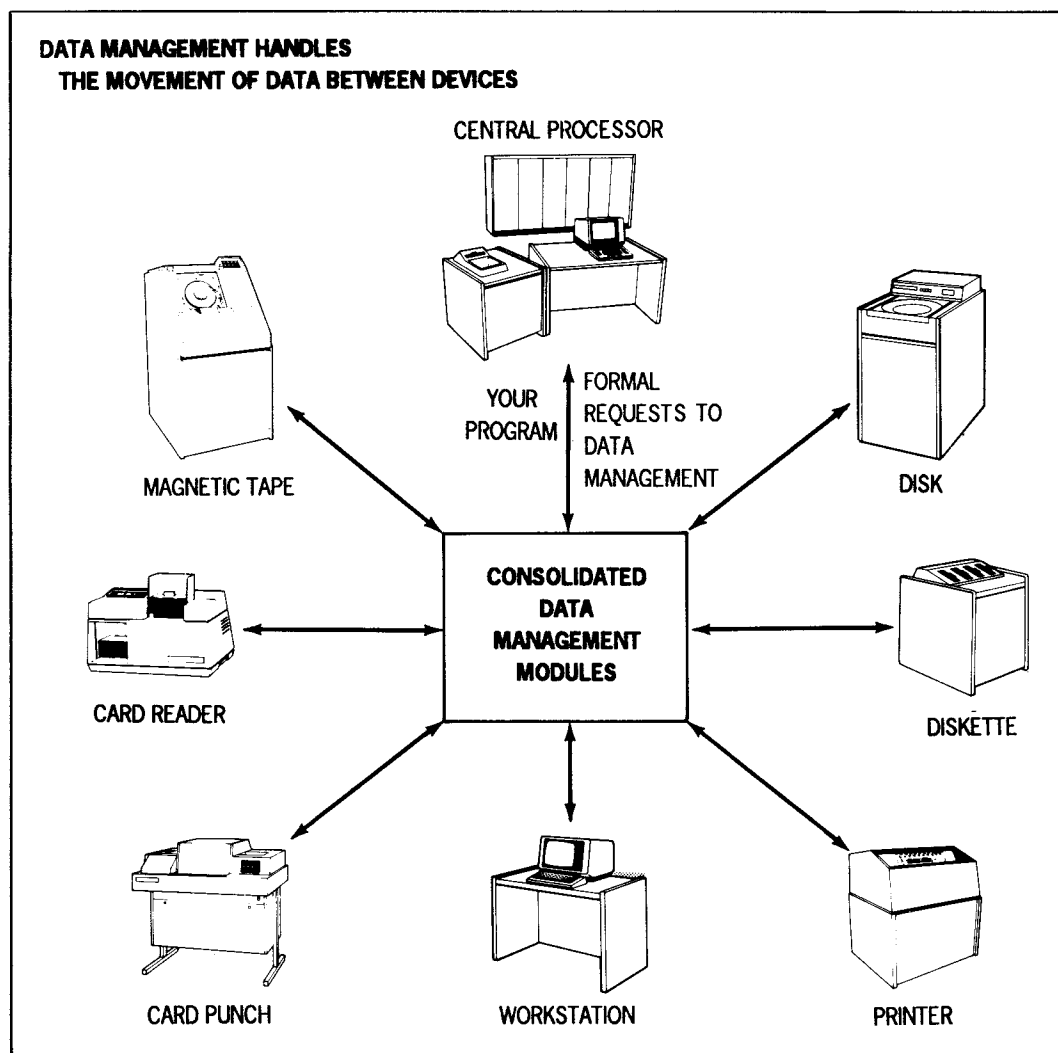
A catalog manipulation utility is provided for the system administrator to perform certain maintenance functions on the catalog. This utility can do the following:

- Assign a protective password to the entire catalog file to prevent unauthorized access
- Make a backup copy of the file and restore it as necessary
- Obtain a listing of the contents of the file

Consolidated Data Management (CDM)

The CDM system is an intermediary 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 workstation data transfers



Logical Input/Output Control System

The logical input/output control system 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

Using 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.

Diskette Access Method

CDM allows accessing records of a diskette file in three ways.

- Sequentially in order of placement
- Randomly by relative record number
- By data set labels

Workstation/Terminal Access Method

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

- Automatic support for the workstation/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 workstation/terminal and user job

- Console-like capability between the user program, system console operator, and workstation/terminal operators
- Support of workstation/terminal function key capabilities
- Device independence

The workstation/terminal access method is enabled automatically. You can also control it through a set of declarative and imperative macroinstructions that connect workstations 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, card readers, card punches, and printers. Records are handled from the first to last according to physical placement. CDM provides sequential access method modules to handle the access requests of user programs for these four device types.

Device Independence

Device independence allows logical data transfers among devices having dissimilar physical characteristics. A single system control table format controls and defines all file types (including disk, diskette, and printer). The files in a program can change from one type to another (for example, disk to diskette) without requiring you to alter the source code of the program that accesses that file. Job control statement changes that associate the file with a device are required. Device-unique commands (for example, tape position commands) should be removed or altered accordingly.

System Service Programs

OS/3 provides a number of system service programs including

- System librarians
- Linkage editor
- Disk, diskette, and tape initialization routines
- Dump routines
- Catalog manipulation utility
- Disk dump/restore
- Disk copy
- 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. Library functions include storing, correcting, adding, duplicating, and transposing library modules. The librarian can modify existing libraries, create new libraries, and duplicate or purge libraries in their entirety. It also ensures that a given library does not contain any modules with identical names and types.
- **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 card, diskette, or 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 or card punch 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 all or selected modules from one library to another
- Deletes selected modules from a 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

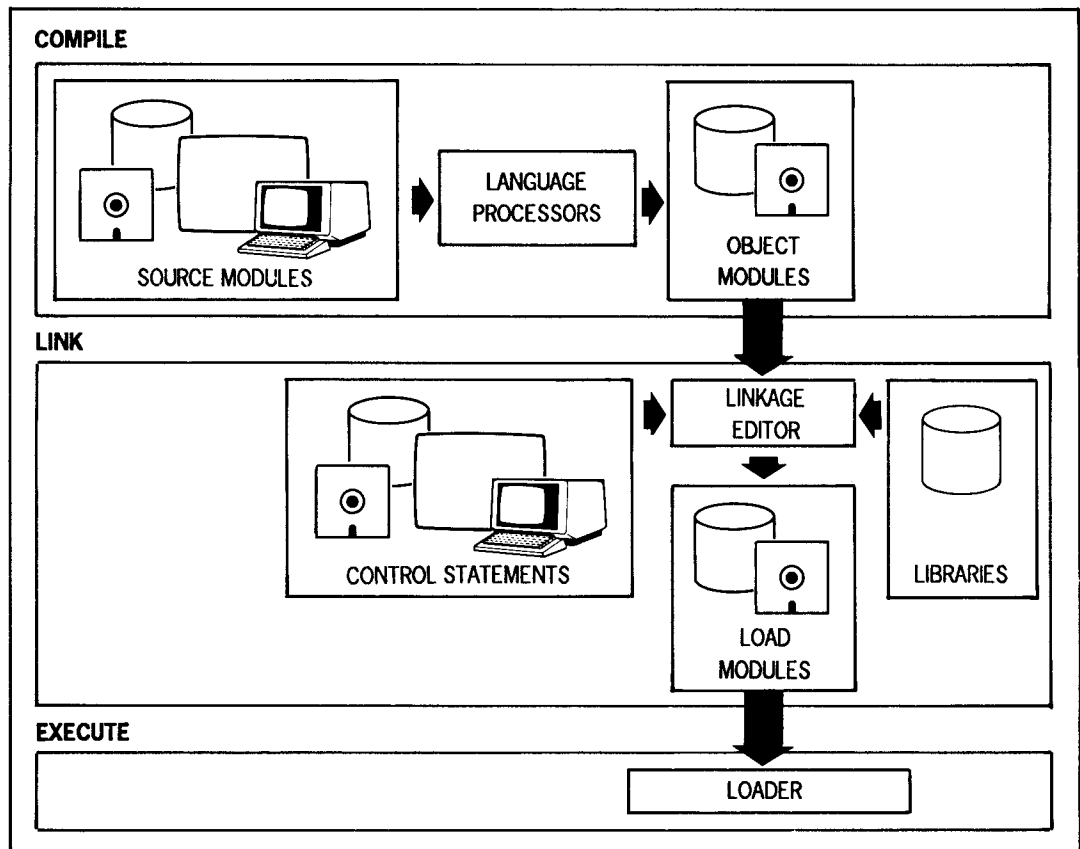
The MIRAM librarian responds to a set of user control statements inserted into the control stream that executed the librarian.

The librarians also 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.

The role of the linkage editor in program preparation is shown in the following illustration.



Disk, Diskette, and Tape Initialization Routines

The various tape, disk, and diskette initialization, or prep, routines check the condition of the magnetic storage media and prepare them for use by the system. The disk and diskette prep routines respond to a set of keyword parameters inserted into the job control stream. These parameters handle expiration date validation, data file allocation, and diskette fast prep. They also assign alternate track, defect skipping, and surface analysis functions.

Dump Routines

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

- **JOB_DUMP** – 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 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. The sections are clearly labeled with headings.
- **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.

Disk Dump/Restore

The disk dump/restore utility lets you make backup copies of the contents of a disk volume. 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 workstation 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.

You use the stand-alone dump/restore program during system installation to read diskettes created in file mode by the standard dump/restore program and write tracks of data to disks. The program can also copy the volume table of contents (VTOC).

Disk Copy

The disk copy utility copies up to six volumes with the option to verify the data copied.

Catalog Manipulation Utility

The catalog manipulation utility maintains the system catalog file that holds the device assignment sets for often-used user files. If a file device assignment set is placed in the system catalog file, it is accessed thereafter through one job control statement. The catalog manipulation utility lets the system administrator assign a protective password to the catalog file, make backup copies of it, and restore these copies to disk when required. The utility operates in response to a set of command statements embedded in the job control stream that executes the utility.

Software Installation Facilities

Unisys delivers OS/3 software on a set of release diskettes or tapes, or on disk. Software installation involves the transfer of this software to a system disk pack. Once the delivered software has been 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.

To accomplish software installation, Unisys provides installation routines as part of the standard OS/3 release. These routines help you install the

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

System Generation Facilities

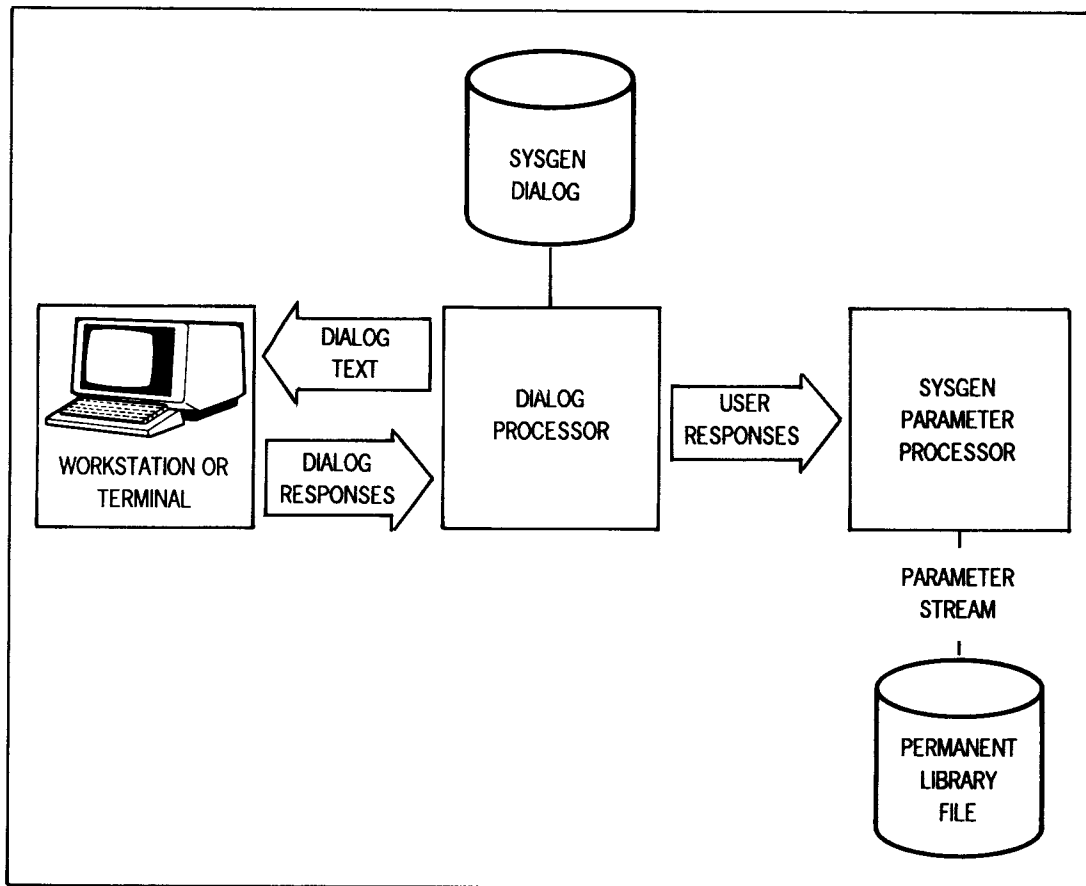
System generation, or SYSGEN, is the process where you define the system's hardware configuration to OS/3 and generate, or create, the control elements required to meet particular processing requirements. To simplify the procedure, Unisys provides the following SYSGEN facilities:

- Dialog
- Parameter processor
- Job control streams

The dialog is an easy-to-use facility to prepare and process required SYSGEN parameters directly through the workstation. You select parameters in response to queries displayed on the workstation screen. 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, in turn, 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 workstation keyins.



Extended System Software

The extended system software provides a number of optional features and improves the utilization and operation of the system. The extended system software is available as a package, and includes the following components:

- Screen format generator (described under Screen Format Services)
- Dialog specification language translator (described under Dialog Processing Services)
- Data utilities
- Spooling and job accounting/reporting facility
- SORT/MERGE
- SORT3

SORT/MERGE and SORT3 are available as separate items.

Data Utilities

The data utilities (DATA) program performs a number of essential data file maintenance and manipulation functions. Through the DATA program, you can

- Make a copy of a data file on the same or different type of storage device
- Reformat the records of a data file
- Select or delete specific areas of a file
- Produce a printed listing of a file in a variety of formats
- Compare duplicated files to their originals to check for discrepancies

The DATA program processes files stored on any of the supported storage media: disk, diskette, tape, or cards. Files can be transferred from one medium to any other medium, and printed listings can be made from files on any medium. You can also perform such device-related functions as label checking, tape rewinding, sequence checking, and writing protected files.

The DATA program executes as an interactive program or runs as a batch job. For interactive execution, you initiate the DATA program through a simple keyin command and enter control specifications in response to a dialog displayed on the screen. Once you have provided all appropriate information to the dialog, the DATA program executes and performs the specified function. If the DATA program executes as a batch job, you must prepare a job control stream. The appropriate DATA program control statements are included in the control stream. The DATA program has the same capabilities regardless of how it executes.

Spooling and Job Accounting/Reporting

Spooling increases the throughput of System 80. Data from low-speed peripheral devices is transferred to disk storage independently of the program that uses the data. When a user program logically retrieves data from a low-speed peripheral, it physically retrieves the data from a higher speed device. On output, the user program logically specifies a low-speed device; but the images are physically recorded on disk storage, and are later transferred to a low-speed device under system control. Spooling of output allows concurrent use of a specified device by multiple programs.

OS/3 spooling consists of the following routines:

- **Input readers** – These accept data files from local or remote subsystems. When these data files are used, the method of submission is not considered.
- **Spooler cooperative** – The spooler routine stores card data in the spool file until the data is required by a user program. The spooler stores data output from a user program in the spool file until a printer or punch is available.
- **Output writers** – Output writers provide local or remote users with output from user programs that receive identical service regardless of final destination.

Spooling is supported for a printerless System 80. It generates printer files for subsequent printing on an OS/3 system that has a physical printer configured. Spool print files can be routed to auxiliary printers. These printers can be connected to an SVT 1122 or UTS 40D local workstation or to a UTS 400, UTS 4000, or SVT terminal. Additionally, spool output can be routed to a PC (using the STEP emulation software) or to your UNIX[®] system.

In a normal spooling operation, job accounting information is generated for each job run on the system. This information is included in the normal message output log; but the information is not saved. However, you can save all the accounting data generated for each job in a spooling LOG file.

The system console and job log accounting utilities gather selected information from the system spool LOG file, sort it according to user specifications, and produce a printed report. The system console and job log accounting utilities are the

- System log accumulation utility
- Joblog report program

The system log accumulation utility transfers all, or selected portions, of the spooling LOG file onto a disk or tape file for use as input to the joblog report program or the user accounting routines. You initiate the system log accumulation routine by running a canned job control stream.

UNIX is a registered trademark of AT&T Information Systems.

Once created, the file can be used by the joblog report program to create a printed report containing selected job accounting information. This program gathers information and sorts it according to user specifications. The joblog report program is also initiated by running a canned job control stream. You select the type of accounting records to include in the report and the manner in which they are sorted.

SORT/MERGE

OS/3 SORT/MERGE operates as an independent sort/merge program or as a modular subroutine. As an independent sort/merge program, defined and initiated by job control, it is constructed as a processor with interrelated modules operating within the framework of a system driver program. As SORT/MERGE executes, the modules required are called by the driver program, loaded into main storage, and executed.

As a modular subroutine integrated into a user program, SORT/MERGE gives you flexibility and freedom. It can be called from a language processor program that includes the verb specifying this operation. The modular structure gives you efficient operation despite variations in hardware configurations and data requirements. It allows a module to be changed, replaced, deleted, or added.

SORT/MERGE CAPABILITIES	
An interface that permits disk or magnetic tape to be used as work area	Input and output that can be associated independently with disk, diskette, or magnetic tape
Sorting of blocked or unblocked records	Sorting of fixed-length or variable-length records
Handling of seven types of key field formats: <ul style="list-style-type: none"> * Character * Binary (signed or unsigned) * EBCDIC data in ASCII collating sequence * Decimal (signed zoned and unsigned zoned) * Leading and trailing sign numeric * Overpunched leading and trailing sign numeric * Floating point (single/double precision) 	Specification of up to 255 key fields
Execution of input and output code	Sorting of noncontiguous key fields in ascending or descending sequence
Shared input and reserved output devices	User specification of alternate collation sequence
A merge-only capability	Sorting of two or more different characters with the same collating value (multicharacter sort)
Convenient restart procedure	Performing of data validity and data integrity checks during sorting
Printing of VOL and LBL information	Automatic extended SORT output file
	Optimized IRAM/MIRAM file handling

IBM Compatible Sort

Unisys also supplies an IBM[®] compatible sort, SORT3. SORT3 is compatible with sort programs provided by IBM for its System/3, System/32, and System/34 data processing systems. SORT3 is an OS/3 program initiated through job control.

SORT3 can process disk, diskette, tape files, or card input, and

- Rearrange the records in a file
- Reformat the records in a file
- Summarize fields in the records
- Select specific records from a file

SORT3 does the following:

- Displays messages at workstations and on the printer
- Prints LBL information
- Provides keyword specifications for month, day, and year
- Optimizes IRAM/MIRAM file handling
- Provides a maximum overflow field length of 256 bytes
- Supports 8470 disks

SORT3 performs three different types of sorts: address out sort, tag sort, and summary sort. The output from the address out sort is a 10-byte (binary) relative record number of the records in the input file. The tag sort output is a file of sorted records that contains control fields and data, control fields only, or data only. The output of the summary sort can be any of the following:

- Control fields only
- Data fields only
- Summary data fields only
- Data fields and summary data fields
- Control fields and summary data fields
- Control fields, data fields, and summary data fields

IBM is a registered trademark of International Business Machines Corporation.

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 consist 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 workstation or 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 provide online HELP screens that provide supplemental information regarding a menu option. The following illustration 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 extensive 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 to 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.
- **Multijobbing** – System 80 supports 48 job slots, of which 47 are available for user jobs and 1 for IS.

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

- General editor
- Screen format services
- Dialog processing services

Workstations

Unisys offers a wide range of workstations with System 80. A typical workstation consists of a keyboard for inputting entries and a video screen for displaying input entries and system responses. (See Section 3 for descriptions of the available workstations.)

You can issue a set of interactive commands directly from the workstation 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 the workstation as an input/output device dedicated to a job or system component or as a program development tool using various interactive facilities, such as the general editor, 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)

ESCORT is a trademark of Unisys Corporation.

General Editor

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 workstation
- Interactively create and update source programs, job control streams, and data files
- Copy, delete, and concatenate files
- Create and call procedures

The general editor 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 ten subfiles: a main work file and nine 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 74 editor, provide syntax checking on source entry for the RPG and COBOL 74 languages. Source entry is through the general editor. Source entry without syntax checking is available directly through EDT screen mode.

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

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

EDT 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. After each line is created or edited, you transmit it to the editor, which places it in a work file created for that session of editorial use. Lines of material, displayed one at a time, roll up the screen as more lines are created or edited.
- **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. You can also use a free-form screen display that provides tabs, which are set for uniform entry of data or used to code assembly language programs. This display also lets you create data files without running a user or system program, simplifying the process of bulk data entry.

EDT offers another aid to creating programs, the error file processor (EFP). The EFP lets you display source program errors on the workstation screen immediately after compiling a program. You can use EDT to correct errors without waiting for a printout of the compiler error listing.

The EFP reads an error file during program compilation. It also reads the file where your source program is stored. It then displays each error message with the line or lines of source code where the error was found. You can see mistakes and quickly correct them by using EDT.

Screen Format Services

OS/3 screen format services simplify variable data entry through standardized data input formats. Screen format services let users

- 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 workstation or terminal for informational purposes

Screen format services consist 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.

Screen Format Generator

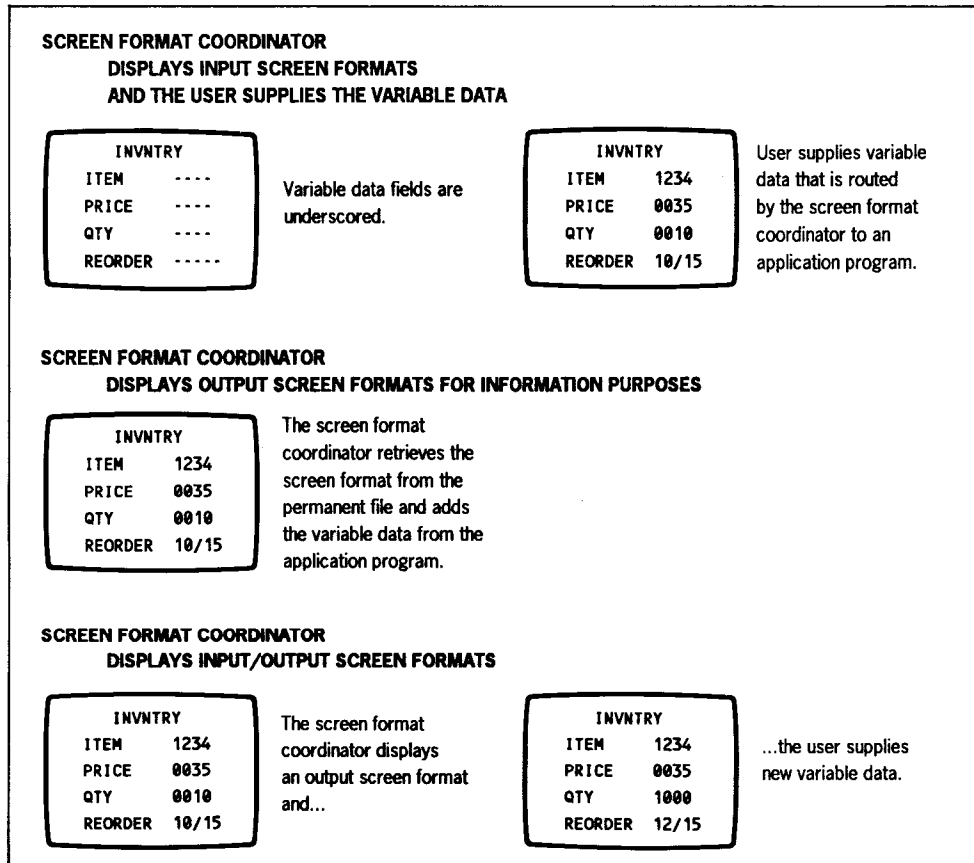
The screen format generator lets you interactively create template-like formats on workstation/terminal screens. These formats simplify the process of entering variable data and parameters, and they present formatted output data on the screen. The screen format generator prompts you for information and provides HELP screens to simplify creation and modification of screen formats.

Once a screen format is created, the generator automatically stores it in a permanent library file for access later by programs that need it or by workstation users for further modification. You can modify an existing screen format to create a new one. Error detection and help capabilities are provided during the creation and modification process.

Screen Format Coordinator

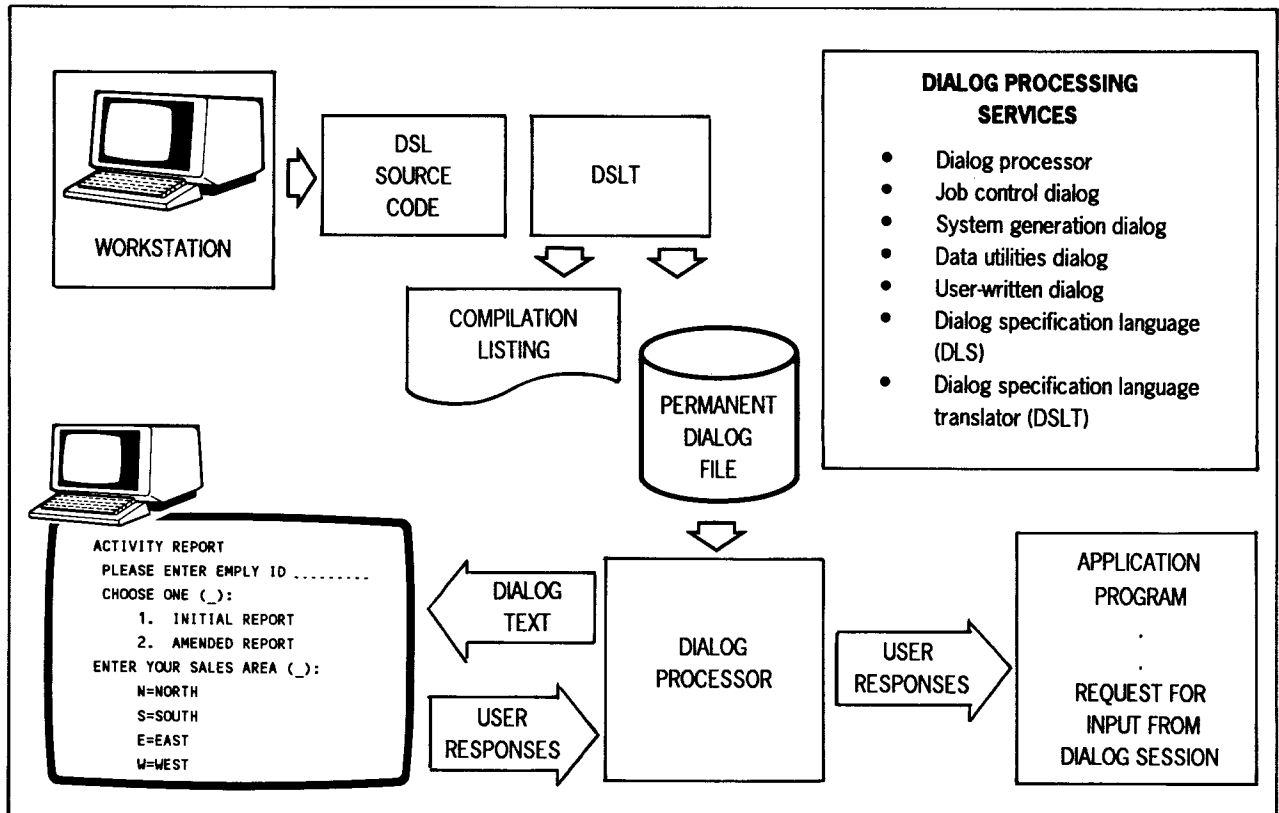
Screen formats are forms displayed on the workstation/terminal screen. You design the formats and a program uses them to input data from or output data to a workstation user. Logically, a screen format is made up of rules used to display and then verify data the operator has entered. You can define numerous screen and field attributes to facilitate end-user requirements. You can request a screen format through selected Interactive Services commands, an IMS action program, an application program, and 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 workstation/terminal user, and routes that data back to the application program that called the screen format coordinator.



Dialog Processing Services

OS/3 dialog processing services let 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 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.



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

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

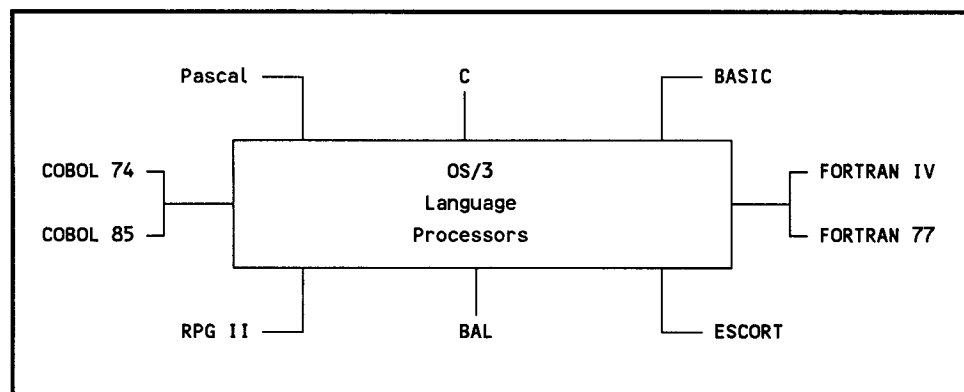
User-written dialogs are written in dialog specification language (DSL). DSL source code is submitted to the dialog specification language translator, which compiles the source code to produce the desired dialog, store the dialog in a permanent file, and produce a compilation listing.

When a program contains 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 OS/3 programming languages.

Enhanced Programming Languages

System 80 features a full range of standard, versatile, programming languages.



COBOL 74 and COBOL 85

COBOL is a general-purpose, yet powerful, language designed for business applications, such as payroll, accounting, inventory, and personnel management. The language is similar to the English language, rather than a form of notation that considers the technical aspects of a particular data processing system.

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 the IMS and the DMS.

The COBOL 74 compiler conforms to the specifications of the *American National Standard COBOL (X2.23-1974)*. 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 1989-1985)*.

COBOL 85 is an enhanced and modified version of the previous COBOL standard (*ANSI X3.23-1974*) with 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 function key 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 applications 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
- Multiexit 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 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 workstation 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 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 capability.

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 workstation 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.
- Workstations 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. Also, indicators, set on when the workstation 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 workstations and workstation error processing.
- The COPY statement includes RPG statements from a library.

Basic Assembly Language (BAL)

BAL is a flexible, machine-level language that offers a variety of features that let 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 specific 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 high-level, interactive programming language generates file processing, data handling programs, and reports. It is easy to use, requires minimal training, yet has the power of more complex data handling languages. The 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 workstation; issues prompting messages to a workstation for input data; and validates the input data. Computational operations available 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 workstation. 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 ESCORT menu screens
- To 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 workstation.

The ESCORT report system lets you create, modify, display, and print reports. The functions of the ESCORT report system are

- Multiple formats for displaying data
- Searching and updating data
- Entering and saving data
- User-defined headers and titles
- Totaling, subtotaling, and averaging data
- Arithmetic computations on each line of data

Here is a typical ESCORT screen display.

```
CHANGE DATA STATEMENT

CHANGE DATA OF file-m (structure-m) (FROM { file-i (structure-i)
                                           WS (structure-i)
                                           WS (structure-i),formname})

FROM clause, structure-names and formname ARE OPTIONAL PARAMETERS
file-m IS NAME OF DATA FILE BEING CHANGED
file-i IS NAME OF FILE CONTAINING CHANGE CRITERIA
IF WORKSTATION IS USED AS CHANGE CRITERIA DEVICE, THEN
  KEYIN WS FOR file-i parameter

KEYIN PARAMETERS
file-m----- structure-m-----
file-i----- structure-i----- formname-----
```

Integrated Communications Access Method (ICAM)

The integrated communications access method (ICAM) is designed for the casual as well as the professional 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 is available for workstation support as well as for terminals supported over a communications line through the ICAM terminal support facility.

ICAM supports the following types of communication:

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

An ICAM configuration includes

- Software modules in the system during SYSGEN
- Communications lines
- Terminal devices
- System utilities that the ICAM network requires
- Programs that interface with the network

ICAM Features

- Message queuing
- Multiple destination routing
- Activity scheduling and priority control
- Timer service
- Checkpoint/restart
- Journal control
- Statistics accumulation
- Trace facility

- Dynamic buffer expansion
- Autodialing
- Public data network support

ICAM Components

- Channel control routine – provides the physical I/O interface to SLCAs and the specific type of communications subsystems.
- Remote device handlers – provide the software logic and control that interface the unique characteristics of specific remote devices to other ICAM components. They are available to accommodate a wide range of terminal equipment, including
 - Local (for IMS) and remote System 80 workstations
 - UTS 20, UTS 30, UTS 40, and UTS 400 Universal Terminal System
 - SVT 1120/1123/1124 terminals and workstations
 - Unisys PCs
 - MAPPER 5
 - IBM 3270
 - UNIX (5000) UNISCOPE[®] Emulator
 - Teletype[®] teletypewriter models 33, 35, 37 and 38

Bisynchronous communications (BSC) 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.
- 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 it provides data throttling and recovery services.

UNISCOPE is a registered trademark of Unisys Corporation.

Teletype is a registered trademark of the Teletype Corporation.

ICAM DCA TS supports

- UTS 20, UTS 30, and UTS 40 terminals
- SVT 1120/1123/1124 terminals
- Unisys PCs
- UNIX (5000) UNISCOPE Emulator

These devices can be used as terminals connected to OS/3 or to an ICAM end user.

ICAM also supports the DCP 15, DCP 30, DCP 40, and DCP 50 running Telcon as remote DCA termination systems, allowing you to function in several modes. The DCP can be connected to the System 80 by a selector channel or 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 the 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, and IMS. 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 the 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 releases in a global environment.

ICAM Interfaces

In addition to providing a communications interface between a terminal network and System 80, ICAM also supports the following interfaces:

- Standard message control program interface
- Transaction control interface
- Demand mode interface
- RPG II telecommunications
- COBOL 74 message control system
- Remote batch processing
- UNIX remote job entry
- OS/3-to-UNIX passthrough mode support
- IBM 3270 emulator
- Remote terminal processor (RTP)

RTP simulates an IBM multileaving workstation by using the Houston automatic spooling program (HASP) protocol through BSC facilities.

RTP interfaces with the following IBM software systems:

- JES2
- JES3
- ASP
- HASP

Nine Thousand Remote (NTR) System Utility

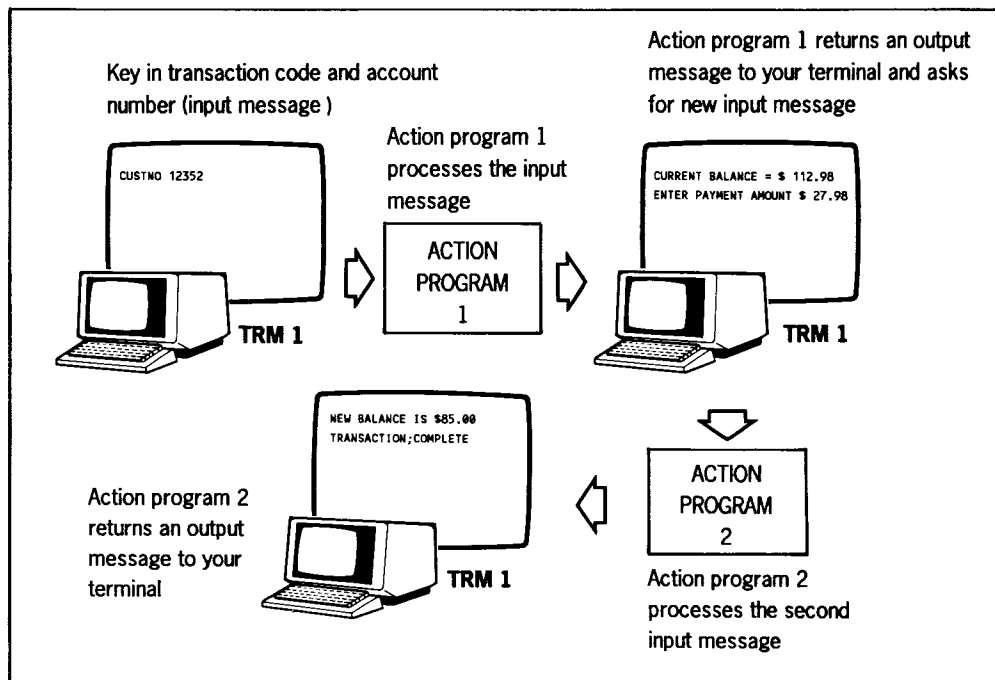
The NTR system utility makes System 80 act as a remote terminal to a Unisys Series 1100 system. NTR is a combination system utility and job task that can run concurrently with other job tasks. It uses a modified version of the direct data interface. NTR provides data transmission control, operator-to-operator communications, and control features for status and command functions between both systems. The NTR utility controls local input and output tasks through a combination of data management macroinstructions and a set of special NTR macroinstructions for user code control of device buffers.

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.



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.

Data File Types

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

- 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.
- 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 CDM mode.

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.

IMS Interface Capabilities

- IMS/CDM – IMS in 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 multithread can be configured to support CDM or data transfer facility.

- **IMS/ICAM** – ICAM presentation services is a group of modules that builds a chain of linked buffers for holding messages at the ICAM remote workstation. This enables ICAM to use its resources more efficiently. Use this facility when you have a remote workstation with a data mode session to IMS.
- **IMS/DMS** – IMS action programs can concurrently access DMS data bases through data manipulation statements embedded in COBOL action programs. DMS modules support 48 job slots.
- **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.

Advanced IMS Capabilities

- **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 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.
- **Multijobbing** - IMS can run in a multijobbing environment because IMS executes as a program under control of OS/3.
- **Dynamic storage allocation** - 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 (DBA)
 - 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

DMS consists of the following components:

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

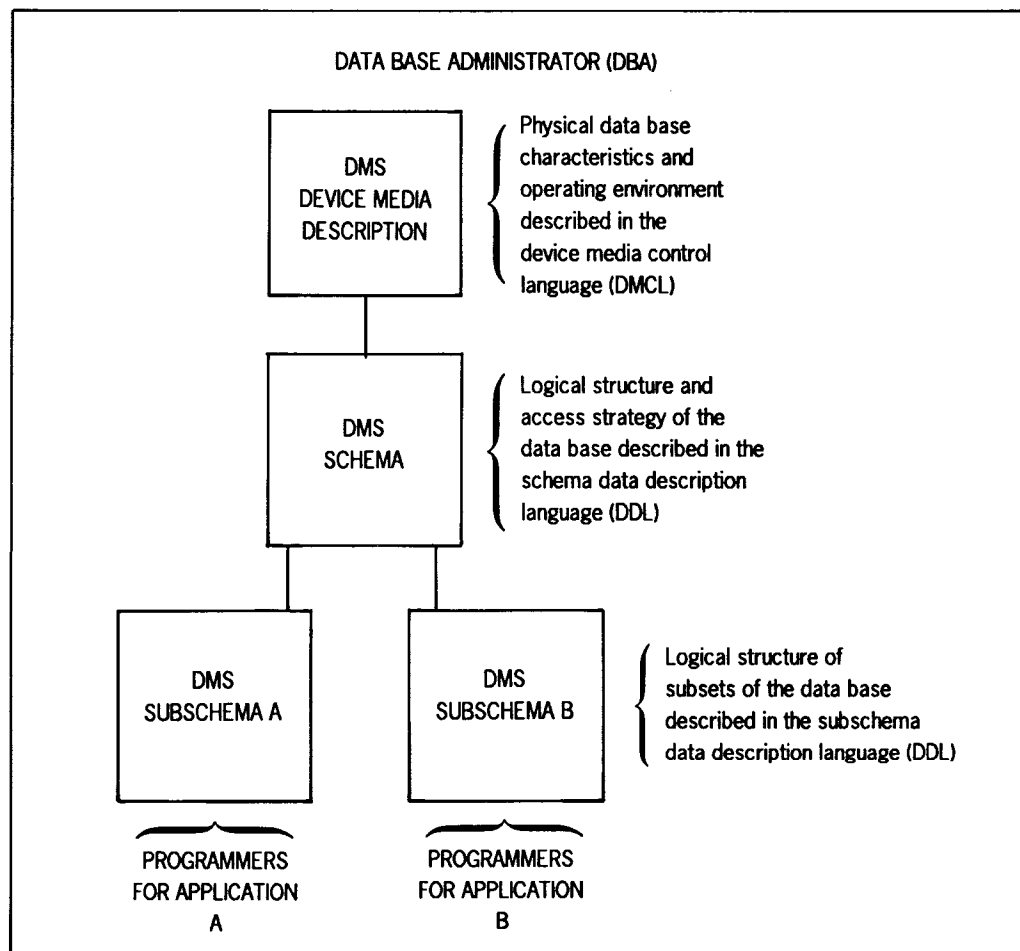
Data Base Languages

There are two data base languages:

- 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 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 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 done in a high-level language (data description language) which is comparable to the data declaration language of COBOL.

The illustration below is a user's view of a DMS data base.



The data dictionary is the data base used by DMS language processors. Two languages access the data base.

- Data manipulation language (DML) – statements are included within conventional COBOL Procedure Division statements for batch processing.
- Conversational data base manipulation language (CDML) – assists the DMS user in accessing a data base. The programmer does not need to prepare, preprocess, compile, or link programs; it is all done by CDML.

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.
- DMS can restore a MIRAM file that is accidentally initialized.
- 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 distributed data processing 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 these 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 and punch spooled 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

The job transfer manager provides commands to

- Submit jobs to the DDP network, monitor their execution, and cancel them
- Communicate with a remote operator's console
- Issue instructions to a remote operating system and respond to messages issued by a job executing on a remote system

File Transfer Manager

The file transfer manager permits you to

- Generate a file directory to catalog the characteristics of files
- Transfer copies of data files and program libraries
- Delete files from the file directory
- Obtain a listing of a file's characteristics from the file directory

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. It 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 lets you perform inquiries and updates on files residing at remote workstations anywhere within the DDP network.

IMS-DDP software requires a configured communications network to physically link the DDP processors at each included 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 workstation.

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 action programs or UNIQUE 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.

Fourth-Generation Languages

System 80 offers two fourth-generation languages.

- LINC II
- MAPPER[®] System

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 exists to assist as the system is defined. LINC II automatically generates every aspect of the system including data base, transaction management, network management, and the application programs. As a result, analysts can focus on the business problem to be solved rather than on technical implementation details.

MAPPER is a registered trademark and service mark of Unisys Corporation.

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

MAPPER System

The MAPPER system provides an interactive application development environment that is conceptually designed for end-users as well as information systems professionals.

The MAPPER system supplies its own data base management system that provides both ease and power in a multiuser environment. It is organized like filing cabinets, with file drawers and individually labeled reports. Reports can be grouped or separated to provide a balance between shared access and data protection. Individual reports permit access to data for checking, updating, or using in applications.

The MAPPER system provides over 100 flexible functions to facilitate computerized information processing. They can also be grouped into runs to create an entire application invoked with a single command.

You can use the MAPPER system in many ways.

- Store and access information quickly and easily
- Combine entire reports (or portions of them) into one master report
- Display data graphically with a variety of patterns and styles of charts
- Send and receive electronic mail through the system
- Schedule the sending of reminder messages
- Control access to confidential information
- Perform "what if?" analysis to help detect and predict trends
- Integrate a series of functions into a run procedure invoked by a single command

MAPPER 5-to-OS/3 Connectivity

A terminal or local workstation user who is signed on to the OS/3 operating system can access MAPPER 5 data in the same manner as a terminal user who is directly connected to the MAPPER 5 processor.

MAPPER 80

MAPPER 80 generates reports from a DMS data base. Data on the DMS can only be accessed by MAPPER 80 software. However, MAPPER 80 utilities can capture MAPPER 80 data from external files (spool or other OS/3 data files) and transfer report data to DMS files. The MAPPER 80 system also uses screen format services. Some available screen characteristics are lowercase translation, original and overlay formats, special editing characters, and nondisplayed fields.

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 four utilities to connect PCs to System 80.

- File transfer (PCTRAN)
- PC online transfer (OLTRAN)
- PC online extract
- PC online disk

PCTRAN

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

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.

OLTRAN

OLTRAN 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 Online Extract

PC online extract provides a bridge between MAPPER software on the host system and applications on the PC. You use a simple MAPPER system report- or spreadsheet-style interface to specify extraction of data from Series 1100, System 11, MAPPER 10, or MAPPER 5 system reports. A PC configured as a System 80 workstation can extract data from MAPPER 5 system reports in OS/3 pass-through mode. You can transfer extracted data to a PC for use in PC spreadsheets, data management, word processing, or other applications.

PC Online Disk

PC online 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.

Maintenance and Diagnostic Software

System 80 hardware diagnostics isolate and identify system hardware faults. The hardware diagnostic routines operate in an interactive programming environment and use displayed dialogs and messages for initiation and entry of control information. The hardware diagnostic system consists of

- Resident diagnostics
- Offline diagnostics
- Online diagnostics

Resident Diagnostics

The resident diagnostic routines are an integral part of the system and perform basic error checking on the central processor complex. The central processor complex includes the central processor, control storage, and the resident disk control hardware. Resident diagnostics also check the paper peripheral, diskette, and workstation controllers along with the single line communications adapter (SLCA). These checks are performed automatically whenever the system is powered on or reset.

MS-DOS is a registered trademark of Microsoft Corporation.

Offline Diagnostics

The offline diagnostic routines consist of microdiagnostic and macrodiagnostic routines. The microdiagnostic routines isolate detected hardware faults to the particular component responsible. Initiated at the console, these routines diagnose problems in the basic processing unit, main storage, and channels. The macrodiagnostic routines check the operability of all system instructions and verify the OS/3 operating system load paths.

Online Diagnostics

The online diagnostic routines check all peripheral and communications devices to ensure proper functioning and operation. These routines are under control of the operating system and run concurrently with user jobs. During normal system operation, errors are logged to allow interrogation and analysis by these routines. Error log edit and analysis programs display individual error events and summary information.

System Maintenance Packages

System maintenance packages (SMPs) contain a collection of software changes that are issued to maintain the system's effectiveness. You install SMPs to maintain and enhance System 80.

Unisys regularly distributes SMPs on diskette. The SMPs contain routines that simplify installation of the SMP and protect system integrity during the SMP application. For example, if an SMP affects installed modules, the system is automatically regenerated to ensure proper operation.

System Activity Monitor (SAM)

SAM measures system efficiency by monitoring and recording various system activities. It is an optional software feature configured during system generation.

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

File Placement Analyzer (FIPLAN)

FIPLAN can improve system performance by projecting optimum file allocation. The projections are based on the contents of the volume's 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.

UTS 400/4000 Support

OS/3 offers the following software components to support the Universal Terminal System 400 (UTS 400) and Universal Terminal System 4000 (UTS 4000):

- UTS 400 COBOL
- UTS 400 edit processor
- UTS 400 load/dump facility

UTS 400 COBOL

UTS 400 COBOL is a high-level, business-oriented international language with features to complement the capabilities of the UTS 400 and UTS 4000 terminals. The UTS 400 COBOL compiler meets the ANSI X3.23-1974 standards and the ISO recommendations on COBOL. It also contains extensions to provide for interactive data entry, program control, and screen management. The UTS 400 COBOL compiler executes under control of OS/3 and produces a compiled program that can be downline loaded to the terminal or placed on a diskette.

UTS 400 Edit Processor

The UTS 400 edit processor lets you create and manipulate text data on a diskette file. It provides an easy and efficient way to create and update line-oriented files of data. You can insert, replace, delete, or change lines in any order. Error messages advise you of mistakes made in entering data. You can insert lines up to 118 characters long in an edit processor file.

UTS 400 Load/Dump Facility

The UTS 400 load/dump facility provides the following facilities for the use of the UTS 400 COBOL and edit processor.

- A communications program to extract a load module from an OS/3 library and transmit this module (downline load) to a remote UTS terminal.
- A communications program to accept UTS 400 dump information and store it on an OS/3 dump file.

Multijob Processing

OS/3 can concurrently process from 1 to 48 jobs in both batch and interactive processing environments. Each job consists of one or more job steps, such as a program compilation or execution of a system-supplied program. The interactive facilities provide maximum response time to each individual workstation. Processor time is allocated based on a system switch list that contains information regarding program priorities, task synchronization, and input/output utilization. While one task is awaiting completion of an external event (such as an I/O request), OS/3 activates another task that is ready to execute. This ensures optimum use of the processor's capabilities and maximizes system productivity.

Batch Job Processing

In batch processing, you build a control stream for a job by coding and keypunching the stream on cards or entering the data on a diskette and submitting the job to the system operator to run. You communicate with the system operator through job control statements in the control stream.

Job control specifications control the way the job is processed. Jobs can also initiate from the control stream of another job through a special job control statement. System console commands store job control streams and programs. You see the results of the batch jobs after the run, sometimes hours or days later.

Interactive Job Processing

In interactive processing, you communicate directly with the system from a workstation, and control all processing activities from program source code creation through the execution of the job. You can request the status of a job; terminate a job; or create, modify, and store programs and control streams in permanent files using the general editor and job control dialog.

Interactive job control facilities let you perform dynamic skip functions to actually change the way a job executes and to submit variable values for a job at run time.

Dynamic Resource Management

OS/3 provides dynamic resource management for both batch and interactive programming activities. You do not need to predict the extent of the resources required for a job and the constraints of the physical devices and software components. The system can dynamically manage

- Main storage assigned to a job
- Unassigned main storage
- Workstations
- Allocation of peripheral devices, such as disks, diskettes, and tape drives

Resource Management Product

In addition to the dynamic resource management OS/3 software routinely provides, Unisys markets a resource management product. This software lets the operator balance job turnaround time and workstation response time by setting the limits for main storage and job initiation without system regeneration. Resource management lets the console operator do the following:

- Control the percentage of main storage for
 - Symbionts
 - Interactive jobs
 - Batch jobs
- Control the maximum number of
 - Total jobs
 - Jobs initiated from workstations
 - Jobs initiated from any single workstation
 - Logged-on interactive users
 - Concurrently executing batch tasks initiated by the ENTER command
 - Concurrently executing run symbionts

The console operator can issue these commands at any time during a session.

Section 3

Hardware

The System 80 processor complex, in either its minimum configuration or with maximum expansion, comprises field-installable modules, including basic processing and I/O control components. High-speed technology is used throughout the system based upon VLSI circuitry. The result is faster processing of larger amounts of data in a more stable computing environment.

Central Processor

The central processor is the controlling center, providing the data path and control logic for instruction execution, system control, and I/O support. It provides a generalized microprogrammed architecture implemented in the following modular system components:

- Basic processing unit with control storage
- System control processor
- Mainframe interface adapter
- Main storage unit
- Instruction cache and operand cache (model 20 only)
- Floating-point processor (model 20 only)

Basic Processing Unit (BPU)

The BPU provides the data path, registers, and associated control logic required for instruction execution and system control. It interfaces to a common bus that provides the common path for all channels and the BPU to access main storage.

The basic BPU contains a four-stage pipeline with a multi-instruction buffer. The BPU for the model 15 is enhanced through micro-accelerator techniques. The BPU for the model 20 is enhanced with an instruction cache, an operand cache, and a floating-point processor.

System Control Processor

The system control processor provides a system function and a service function. The primary system function is system clock generation. The service function is provided by a service processor that supports diagnostics, maintenance, service frames, console support, initial program load, and initial microprogram load for the system.

Mainframe Interface Adapter

The mainframe interface adapter is the interface between the IOP channel or channels and the main system bus. It also includes a separate bus for loading IOP microcode and performing IOP maintenance operations.

Main Storage Unit

The main storage unit stores data in a four-byte word format. It checks address and data for correct parity and provides double- and single-bit error correction, as well as data alignment and error correction codes. Storage protection ensures data and program integrity in the system. Main storage ranges from 2 to 8 megabytes in model 10, from 4 to 12 megabytes in model 15, and from 4 to 16 megabytes in model 20. Main storage expansion is accomplished in 2-megabyte and 4-megabyte increments for model 10, and 4-megabyte increments for models 15 and 20 by basic memory unit feature selection. Additional bits are also provided for error detection and correction.

Operand and Instruction Caches

The model 20 is accelerated through a 16K-byte instruction cache and a separate 16K-byte operand cache. The operand cache uses a *store through* method for main storage. Cache bypass is provided for selected instructions.

Floating-Point Processor

The model 20 also receives processing acceleration through the use of a separate floating-point processor. Floating-point and multiply instructions are accelerated.

I/O Facilities

The system provides a complete range of I/O facilities. One input/output processor (IOP) and one selector channel are an integral part of each system. You can use five additional selector channels and one additional IOP.

I/O Processor

The IOP supports low-speed peripheral subsystems through the shared direct memory access function. Paper peripherals, local workstations, diskettes, and tapes can be attached through the IOP. The multiple line communication multiplexer and associated single line communications adapters (SLCAs) are used to connect a comprehensive complex of communications devices and provide interconnection capability to other systems.

Selector Channel

The selector channel is used to control high-speed I/O devices including disks, magnetic tape, high-speed printers, card readers, DCP/15, DCP/30, and DCP/40. Each selector channel can interface up to eight freestanding controllers, or four freestanding controllers plus integrated disk control units (IDCUs) in any combination with no more than three IDCUs. The selector channel operates at 2.2 megabytes per second.

Instruction Set

The instruction set for System 80 consists of general, floating-point, decimal, and system control instructions. Compatibility with the Series 90/30 system and other System 80 systems is maintained on nonprivileged instructions except where restrictions are noted for 90/30.

Nonprivileged Instruction Set

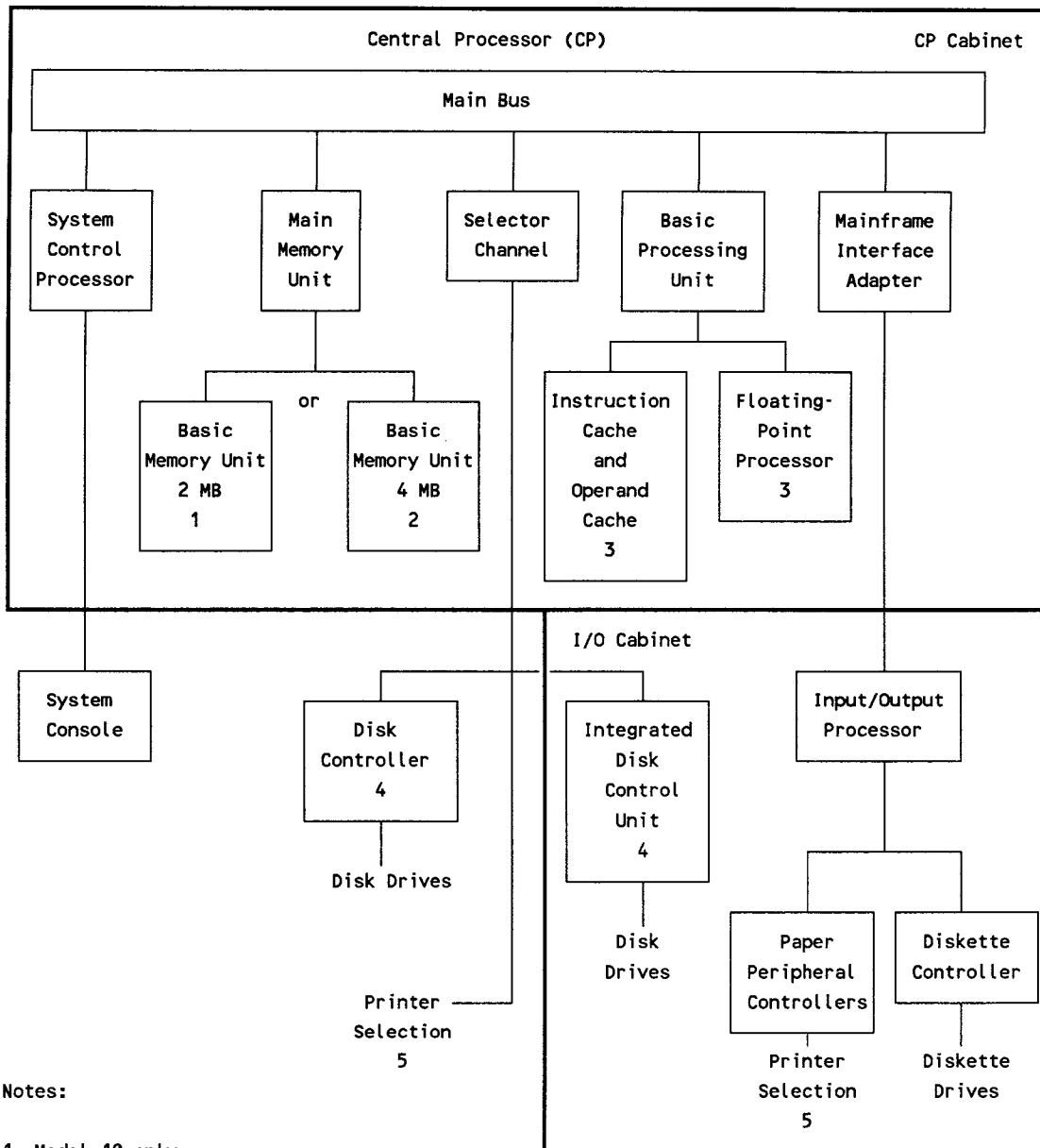
Nonprivileged instructions process fixed-length binary numbers, floating-point numbers, packed and unpacked decimal numbers, and EBCDIC or ASCII characters. You can transfer data between main storage and the user program set of general registers, as well as from one location in main storage to another.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

Privileged Instruction Set

Privileged instructions are used by the operating system software when operating in the supervisory state. In this state, all installed instructions are valid and can be executed. This set of instructions includes facilities to load and store the contents of low-order main storage and to load the writable section of the microprogram control storage. You cannot include privileged instructions in a user program.

Hardware Configurations

The extensive capabilities of System 80 give you an almost unlimited variety of system configurations. The system is configured from the basic processor complex to maximum expansion with two expansion cabinets. The illustration below shows a minimum configuration for System 80.



Notes:

- 1 Model 10 only
- 2 Models 15 and 20 only
- 3 Model 20 only
- 4 Minimum disk configuration 50 MB on one drive or at least two drives of the following types: 8419, 8430, 8433, 8494, 9720.
- 5 Paper peripheral controllers required if line printer is not configured on selector channel.

Basic System

The basic system includes the central processor components and a minimum set of I/O subsystems as follows:

- Processor cabinet and I/O cabinet with power
- Central processor, four-stage pipeline
- Main storage processor and main storage unit with 2 megabytes of main storage for model 10 and 4 megabytes of main storage for models 15 and 20
- One selector channel
- Floating-point processor (model 20 only)
- Operand cache (model 20 only), 16K bytes capacity, 200 ns access
- Instruction cache (model 20 only), 16K bytes capacity, 200 ns access
- I/O processor with a diskette controller and diskette drive
- One disk control unit
- Either one large capacity disk drive or two medium capacity disk drives
- System console with system control panel
- Paper peripheral controller and line printer, or a line printer attached to the selector channel

The console can be located up to 30 feet (9 meters) from the processor cabinet. The diskette subsystem can be located up to 50 feet (15 meters) from the processor or I/O cabinet.

System Growth

You can expand System 80 from its minimum configuration to meet the increasing data processing demands of your growing business. Following are some capsule descriptions of the expansion capabilities of System 80:

- Main storage expansion from 2 to 8 megabytes (model 10), 4 to 12 megabytes (model 15), and from 4 to 16 megabytes(model 20)
- Up to 48 removable and nonremovable media disk drives; up to 11 billion bytes of online disk storage
- Up to 120 locally connected workstations
- High-speed magnetic tape subsystems

- Streaming tape subsystems for backing up nonremovable media disk storage
- Additional paper peripherals, including printers, card readers, and punches
- Communications control units, capable of interfacing with System 80 models 4, 6, and 8 and models 10-20 systems running under OS/3
- Distributed communications processor (DCP) connected through the selector channel

Migration from Other Systems

The System 80 models 10-20 offer growth capability as upgrade and replacement systems for System 80 models 3-6 and model 8. They also provide upgrades for Series 90/25, 30, and 40 systems. Software compatibility with these models complements Unisys base growth commitment.

A System 80 model 10 system can be easily upgraded to a model 15 or 20 system at your location without replacing the basic hardware or changing the software.

Peripheral Subsystems

System 80 can operate with a wide variety of peripheral subsystems, which are illustrated and briefly described in the following pages.

Disk Subsystems

A minimum System 80 configuration requires at least one nonremovable, high-capacity disk drive or two removable, medium-capacity disk drives.

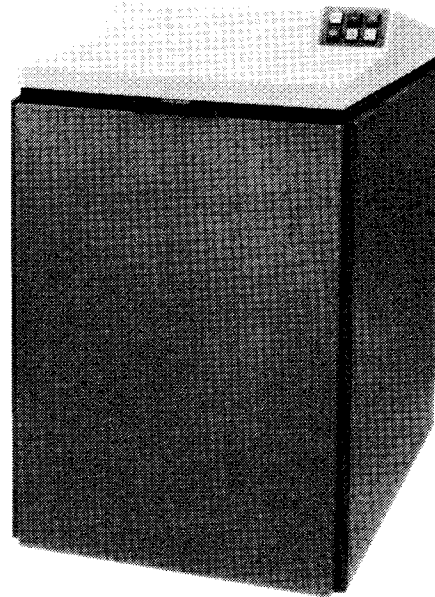
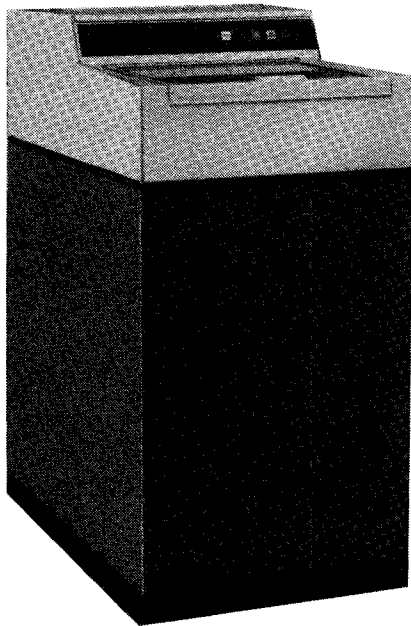
System 80 models 10-20 support the following disk subsystems:

- 8416/8418
- 8430/8433
- 8470/8480*
- 8417*
- 8419
- 8494*
- 9720*

* Fixed media drives

8416/8418 Disk Subsystems

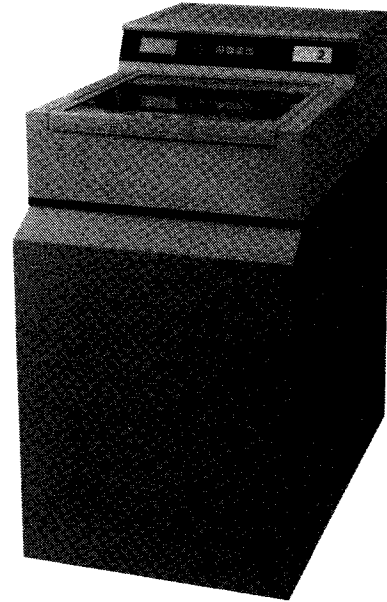
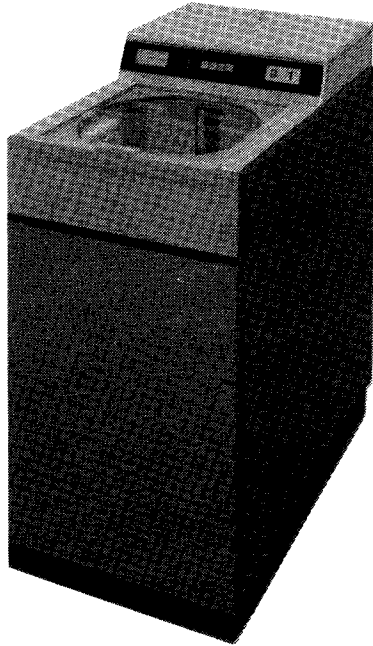
An integrated controller located inside the System 80 models 10-20 processor cabinet controls up to eight 8416 or 8418 disk drives. The drives operate through the integrated disk channel.



	8416	8418
Record format (bytes)	256	256
Tracks per cylinder	7	7
Number of cylinders	404 + alternates	808 + 7 alternates
Data rate	625 KB/s	625 KB/s
Average access time	27 milliseconds	33 milliseconds
Data capacity	29 MB	58 MB
Disk media	Removable	Removable
Rotational speed	2,800 rpm	2,800 rpm

8430/8433 Disk Subsystems

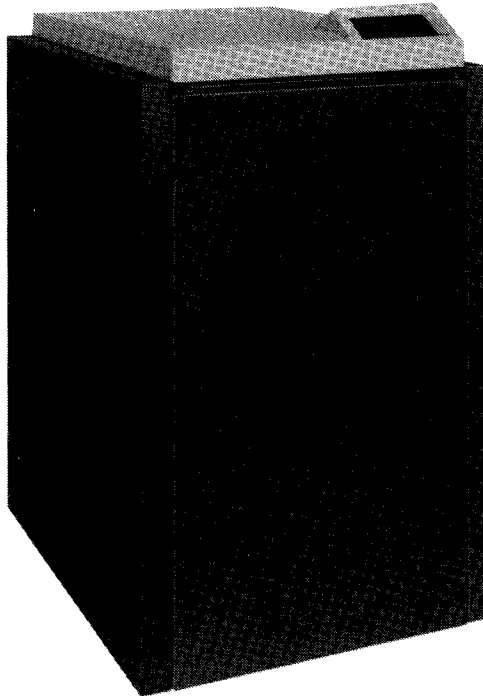
The 8430 and 8433 disk subsystems use their own type 5039 control unit in a freestanding cabinet. The disk subsystem is connected to the system through the selector channel. A single control unit supports up to 16 disk drives in any intermix of types 8430 and 8433. Multiple disk subsystems can be configured and dual-channel operating capability is available.



	8430	8433
Record format (bytes)	Variable	Variable
Tracks per cylinder	19	19
Number of cylinders	411	815
Data rate	806 KB/s	806 KB/s
Average access time	27 milliseconds	30 milliseconds
Data capacity	100 MB	200 MB
Disk media	Removable	Removable
Rotational speed	3,600 rpm	3,600 rpm

8470/8480 Disk Subsystems

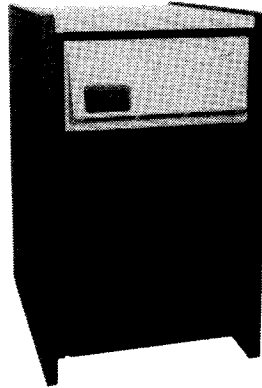
The 8470 disk subsystem uses fixed media disk drives and is controlled by a disk controller located inside the processor cabinet. The 8470 disk subsystem has an optional dual-access capability (by two controllers). You can connect up to eight drives to one controller. The 8480 disk subsystem (not shown) has the same operating characteristics as the 8470 model. It contains four fixed media drives in a single cabinet that requires less floor space than four 8470s. One 8470 subsystem counts as four devices toward the maximum permitted on the IDCU.



Record format (bytes)	1024 (physical); 256 (logical)
Tracks per cylinder	32
Number of cylinders	625 + 5 alternates
Data rate	2.1 MB/s
Average access time	23 milliseconds
Data capacity	491 MB
Disk media	Fixed
Rotational speed	3,600 rpm

8417 Nonremovable Disk Drive

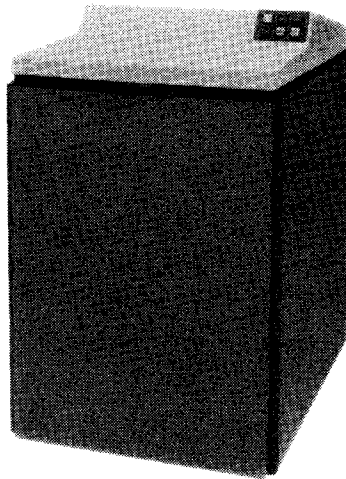
The 8417 nonremovable disk drive is a freestanding disk storage device. It is capable of storing up to 118.2 megabytes of information. Up to three 8417 disk drives can be housed in a single cabinet.



Record format (bytes)	256
Tracks per cylinder	14
Number of cylinders	550 + 10 alternates
Bytes per disk (MB)	118.2
Data rate (MB/s)	1.1
Head positioning time (ms)	
Minimum	7
Average	35
Maximum	70
Rotational latency (ms)	
Average	8.8
Maximum	17.6

8419 Removable Disk Drive

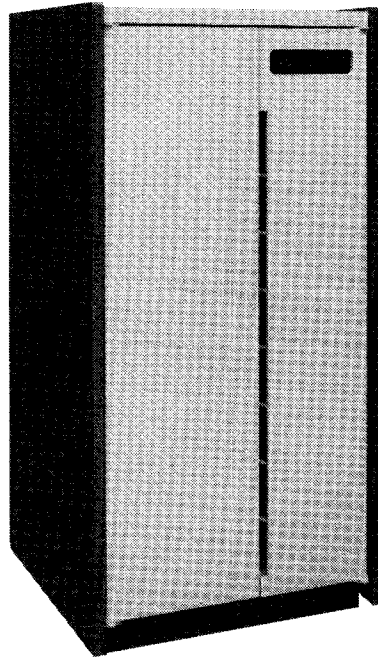
The 8419 removable disk drive is a freestanding disk storage device that includes its own power supply and operator control panel. It is capable of storing up to 72.3 megabytes of information.



Record format (bytes)	256
Tracks per cylinder	7
Number of cylinders	808 + 7 alternates
Bytes per disk (MB)	72.3
Optional fixed-head capacity (MB)	NA
Data rate (MB/s)	0.78
Head positioning time (ms)	
Minimum	10
Average	33
Maximum	60
Rotational latency (ms)	
Average	10.7
Maximum	21.4

8494 Disk Subsystem

The 8494 disk subsystem uses fixed media disk drives. It can be used both as a SYSRES or a data device. The minimum system consists of one control unit and two 300-megabyte devices. A complete subsystem of eight disks and two controllers can be housed in a single cabinet. You can add disk drives in increments of 2.



Record format (bytes)	512 (physical); 256 (logical)
Tracks per cylinder	20
Number of cylinders	626 + 6 alternates
Data rate	2.2 MB/s
Average access time	18 milliseconds
Data capacity	307 MB
Disk media	Fixed
Rotational speed	3,600 rpm

9720 Disk Subsystem

The 9720 disk subsystem uses fixed media disk drives and is connected to System 80 via the selector channel. The minimum system consists of one control unit and one drive module containing two drives. Drive modules can be added in single increments to a maximum of four in a cabinet. A second controller can be added to the basic cabinet. A maximum configuration consisting of two controllers and four drive modules provides a 2x8 system. FIPS 61 power sequencing is optional.

Record format (bytes)	512 (physical); 256 (logical)
Tracks per cylinder	27
Number of cylinders	745
Data rate	3.0 MB/s
Average access time	17 milliseconds
Data capacity	776 MB
Disk media	Fixed
Rotational speed	3,600 rpm

Workstation Subsystems

A workstation subsystem consists of a microprocessor controller and up to eight workstations. You can attach additional workstation controllers to expand the number of workstations in the system to 120 maximum.

Workstation Capabilities

The workstations are designed for ease of operation. They include many operating capabilities for maximum versatility and access to the system. Indicators show you the mode of operation (system or workstation) when power is on. They also tell you when there are messages waiting to be transferred from the system to the workstation.

Local Workstations

Three types of local workstations are available with the System 80.

- Model 1
- Model 2
- SVT 1122

You can intermix these workstations on the same workstation controller. Only the model 1 and model 2 workstations are described here. The SVT 1122 workstation, which is similar to the model 2, is described under "Communications Subsystems."

Each local workstation can be located up to 5,000 cable feet (1500 meters) from the processor cabinet. The controller uses dedicated buffers to allow data transfers at a rate of 19,200 bits per second. The model 2 workstation offers an increased functionality over the model 1 workstation because it is user programmable and offers an expanded selection of peripheral devices. (The SVT 1122, however, cannot be programmed by the user.)

Model 1 Workstation

The model 1 workstation features the following:

- Protected characters – You can protect characters displayed on the screen so that the operator cannot position the cursor on them. If all character locations are protected, the cursor moves to the home position. Characters to be protected are selected by programming commands. Protected characters appear on the display with different intensity or in reverse video.
- Cursor scan – You can scan the cursor backward, forward, up, or down by pressing appropriate keys on the keyboard. You can make rapid corrections or changes by entering the desired character location through the shortest path.

- Operating modes – The workstation operator maintains complete control of the operating environment by allowing the use of console commands to the operating system without restricting use of the display screen. There are two operating modes.
 - Workstation mode – In normal operation, the workstation mode provides communications to application programs.
 - System mode – In system mode, the workstation is directly connected to OS/3. Workstation mode is suspended, and the top two lines of the display indicate system information. The first line is used by the operator to enter inputs to the system; the second line displays the system response messages. Lines 3 through 24 continue to display workstation mode information.
- Blink marker characters – Special characters can be selected for blink (delta) display fields. This helps the operator change the contents of unprotected locations within blink fields. The delta symbols indicate the beginning and end of these fields.
- Reverse video display – Programmers can designate areas of the terminal screen for reverse video display. This highlights a particular point on the screen. Reverse video does not affect other operations of the workstation in either mode.

Model 2 Workstation

The model 2 workstation provides all the features of the model 1 workstation and offers three important additional features:

- Additional peripheral support – A 132-column correspondence-quality printer produces clean, clear-cut characters on business letter head paper or continuously fed and sprocketed multipart forms. The printer operates at a speed of 45 characters per second. When horizontal spacing is set at 12 characters per inch, the printer can print up to 158 characters per line.
- Screen bypass – This feature lets you specify a model 2 workstation with a printer as two separate devices. You can output data to a printer connected to the workstation at the same time the workstation is engaged by other system functions or user programs.
- User programmability – A most important feature of the model 2 workstation (UTS 400 only) is that you can program it for such functions as data creation, formatting and validation, and various arithmetic operations.

Programs for model 2 are coded in a special subset of *American National Standard COBOL X3.23-1974*. They are compiled and linked on the System 80 to which the workstation is connected. The resulting load module is downline loaded into the peripheral diskette or directly into the workstation memory. The model 2 can be configured with up to 64K bytes of storage. Storage is added in increments of 32K bytes.

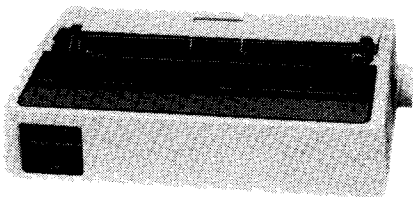
Local Workstation Characteristics

Type of display	Cathode ray tube (CRT), green phosphor display 12-inch screen model 1 and 2, 14-inch screen SVT 1122
Number of display lines	24 data plus 1 status line
Characters per line	80 for model 1 and 2, 80 or 132 for SVT 1122
Keyboard arrangements	Three for model 1 and model 2 - Typewriter layout - Typewriter layout with numeric and function pads - UTS 400 layout SVT 1122 has UTS 30 layout (94 keys)
Direct connect communications interface characteristics	- Serial - Asynchronous, full-duplex connection - 19,200 bps - Optically isolated
Languages supported	Domestic plus the languages from the following: United Kingdom, Germany, France, Spain, Denmark/Norway, Sweden/Finland, and Italy
Number of display screens	One for model 1 Two for model 2 and SVT 1122
Peripheral device support	None for model 1 One for model 2 and SVT 1122

Workstation Printers

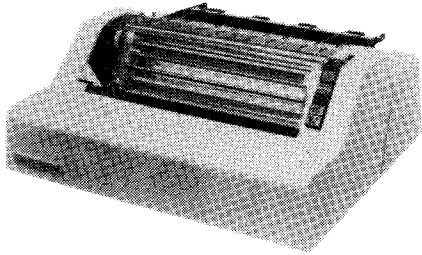
The following is a sample of the printers supported by the model 2 workstation.

80-Column Printer



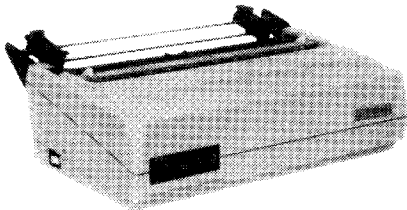
The 80-column printer is an inexpensive printer, lightweight and compact. A tension-loaded printhead forms 7-by-9 dot matrix characters. The printer prints at speeds up to 80 characters per second. The printer is available with a friction-feed platen to print cut-sheets or with a pin-feed platen for printing on continuous-feed forms. (Not available on the SVT 1122.)

132-Column Printer



The 132-column printer maintains the economy of character-by-character serial printing while offering high-speed printing rates of up to 200 characters per second. The same character sets available for the workstation are offered for the 132-character printer. The USA character set is available as a standard 64-character set, a standard 96-character set, and a 96-character set with descenders. The latter includes certain letters (that is, j, p, and q) formatted so that the tail of the letter descends below the normal print line. (Not available on the SVT 1122.)

0425 Printer



The 0425 printer uses a dot matrix printing system to print at speeds of up to 160 characters per second. Two different dot matrices are available with the printer. The standard is a 9-by-7 dot matrix, with lowercase descenders, for draft-quality printing. Available as a special feature is an 18-by-40 dot matrix producing high-resolution characters at speeds of up to 40 characters per second. The printer prints up to 132 columns. Paper can be fed with a friction platen, capable of producing an original and up to three copies; or the printer can use sprocketed forms with a forms tractor, capable of producing an original and up to five copies. Variable spacing is selectable at either 6 or 8 lines per inch. Horizontal spacing can be set at 10, 12.5, or 16.6 characters per inch. (For the SVT 1122, the model 25 printer is the same as the 0425 printer.)

Other printers supported by the model 2 and the SVT 1122 workstation are the following:

Model 2 Printers

Model 425
Model 789
Model 791

SVT 1122 Printers

AP 1327 dot matrix
Model 25
AP 1329 dot matrix

See your local Unisys representative for a detailed description of printers best suited to your workstation requirements.

Remote Workstations

There are seven remote workstations used on the System 80.

- The model 1 remote workstation (UTS 20) is similar to the model 1 local workstation (UTS 20D).
- The model 2 remote workstation (UTS 40) is similar to the model 2 local workstation (UTS 40D).
- The UTS 30 remote workstation is similar to the model 2 remote workstation (UTS 40).
- The SVT 1123 remote workstation is a modification of the SVT 1120 terminal. It provides system response mode and restricts operator setup languages to English, German, French, and Italian.
- The SVT 1124 remote workstation is also a modification of the SVT 1120 terminal. It provides system response mode and restricts operator setup languages to English, German, Spanish, and Dutch.
- The Unisys family of PCs can also be used as remote workstations.
- UNISCOPE emulation on the UNIX 5000 systems also provides system response mode, so it can be used as a remote workstation.

System Console

The system console permits the system operator to power up the system, control its operation, and power it down at the close of business. In addition, it serves as the interface between the system operator and system software, and provides diagnostic information for the operator and maintenance personnel. The system console operates in the following three modes:

- System console mode – provides communications between operator and system software for system console functions.
- System control mode – provides communications between operator and system control functions, such as initial microprogram load (IMPL) or initial program load (IPL).
- Maintenance mode – provides communications between the Unisys customer engineer and system hardware for maintenance panel functions.

Alternate Console

The system operator can define a specific local workstation as an alternate console. Only one alternate console can be generated per supervisor.

Printer Subsystems

System 80 supports

- 0789 line printer
- 9246-14B high-performance line printer
- 0776 high-performance line printer
- 0770 line printer (model 0770-04/05)
- 0770 II line printer (models 0770-06/07 and 9246-25B)
- 0798 character printer

Printer subsystems are connected to the System 80 through the paper peripheral controller and selector channel. Character printers can also be attached to each workstation.

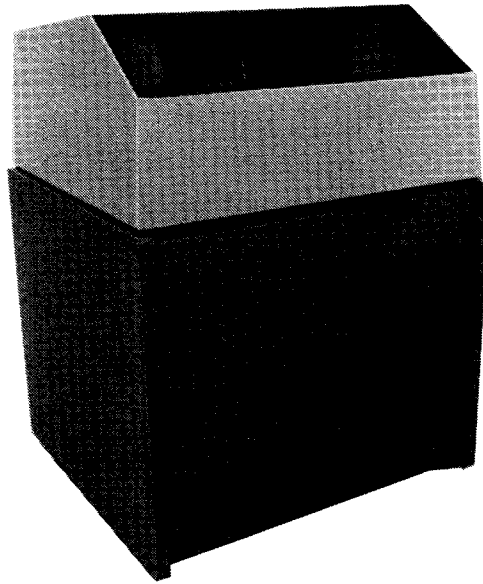
The 0789 printer, 0776 high-performance line printer, and 0798 character printer are connected through the paper peripheral controller. The selector channel supports the 0776, 0770, and 9246-14B printers (high-speed printers).

A remote printer attachment permits the connection of a remote printer, either the 0789 or 0798 line printers, or the 200 cps printer. Print speed does not decrease when printers are connected as remote printers.

0789 Line Printer

The 0789 printer is a freestanding impact line printer. It uses a horizontal moving band in a print cartridge assembly that makes it easy to change print cartridges.

Three printer speeds are available with the 0789 line printer: 180, 300, and 640 lines per minute (lpm). The exact print speed is determined by the print band speed, paper advance time, data transfer rate, the number of times an individual character appears on a print band, and the character density (characters per inch).

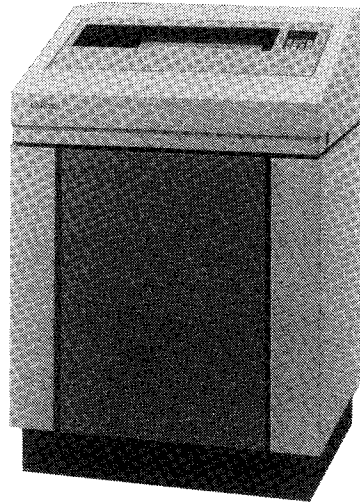


Print speed	180, 300, or 640 lpm for 48-character set
Print line	132 columns wide
Printing density	10 columns per inch
Line spacing	6 or 8 lines per inch (operator selectable)
Line advance	30 ms at 6 lines per inch
Forms slew rate	15 inches per second (38 cm/s)
Forms width	16 inches (41 cm)
Forms length	22 inches (56 cm)
Number of copies	1 original plus five carbons

9246-14B High-Performance Line Printer

The 9246-14B high-performance line printer is a freestanding impact line printer capable of speeds of up to 1,200 lines per minute. Printing begins when the print buffer has been filled and forms advance has been completed.

The type media consists of a flexible metal band that contains etched characters and moves continuously in a horizontal direction scanning the print position.

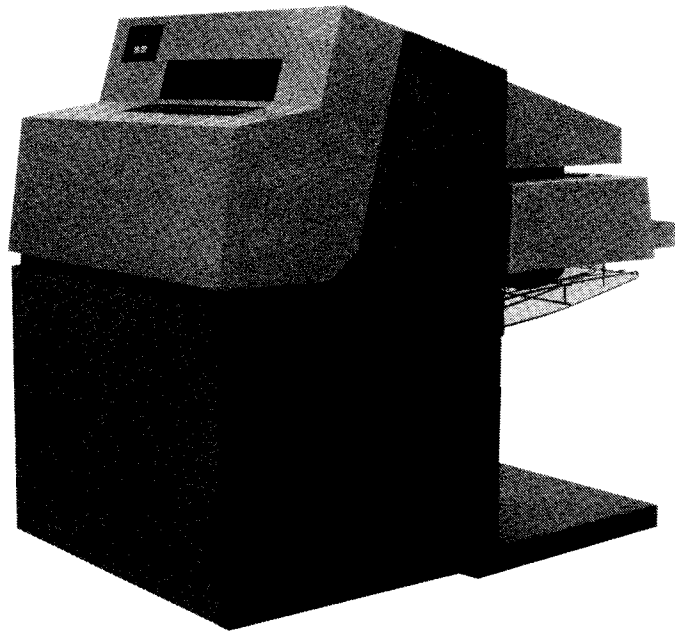


Print speed	1,200 lpm for a 64-character set
Print line	80, 132, or 136 columns wide
Printing density	10 columns per inch
Line spacing	6 or 8 lines per inch (program selectable)
Line advance	14 ms at 6 lines per inch
Forms slew rate	15 inches per second (38 cm/s)
Forms width	3 to 16 inches (7.7 to 41 cm)
Forms length	3 to 14 inches (7.7 to 36 cm)
Number of copies	1 original plus five carbons

0776 High-Performance Line Printer

The 0776 high-performance line printer is a freestanding impact line printer capable of print speeds up to 1,200 lines per minute. It uses a horizontal-moving print band in a print cartridge assembly that makes it easy to change print cartridges.

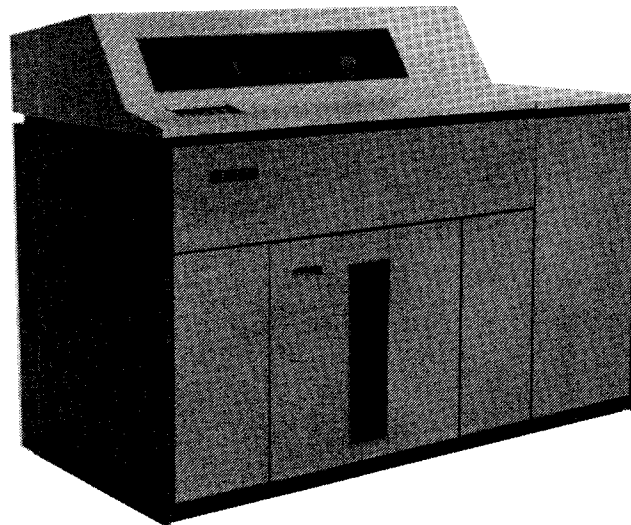
The print rate depends on the model printer and character set used for printing. For example, the print speed of model 0776-00 for a character set containing 48 characters is 760 lines per minute, while the print speed of model 0776-06 for a character set containing 48 characters is 1,200 lines per minute.



Print speeds	760, 900, or 1,200 lpm
Print line	120 columns or 136 columns by feature
Print density	10 columns per inch
Line spacing	6 or 8 lines per inch (operator selectable)
Line advance	16 ms at 6 lines per inch (single line)
Forms slew rate	22 to 50 inches per second (127 cm/s)
Forms width	18.75 inches (47.6 cm)
Forms length	18 inches (45.7 cm)
Number of copies	1 original plus 5 carbons

0770 Line Printer

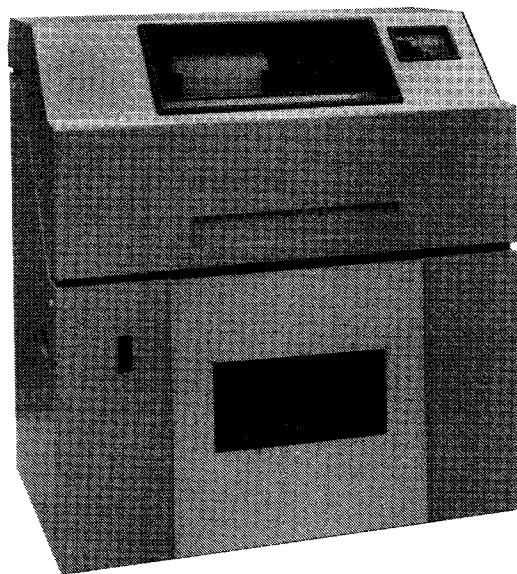
The 0770 printer (model 0770-04/05) is a freestanding impact line printer. It can print at speeds of up to 2,000 lines per minute. The 0770 printer contains its own controller and is connected to the system through the selector channel.



Print speeds	800, 1,400, or 2,000 lpm, depending on model
Print line	132 columns or 160 columns by feature
Print density	10 columns per inch
Line spacing	6 or 8 lines per inch (program selectable)
Forms advance rate	50, 75, and 100 inches per second depending on the model
Data codes	EBCDIC, ASCII, or any code of 6, 7, or 8 bits
Character set	48 standard, up to 384 by feature
Vertical format	Program control
Forms width	3.5 to 18.75 inches (8.89 to 47.6 cm)
Forms length	7 to 24 inches (17.8 to 61 cm)
Number of copies	1 original plus 5 carbons

0770 II Line Printer

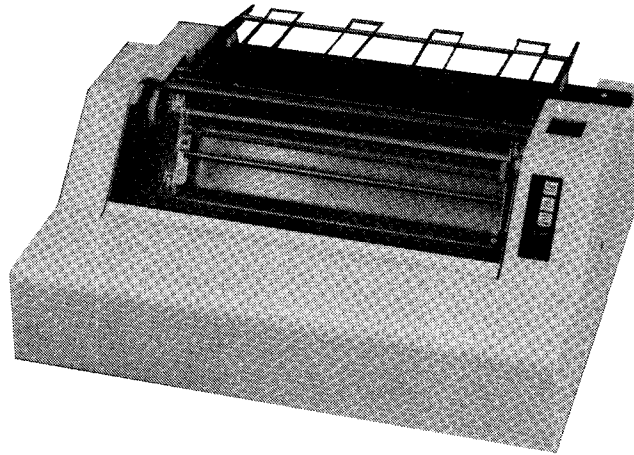
The 0770 II printer (models 0770-06/07 and 9246-25B) is a freestanding impact line printer. It can print at speeds of up to 2,000 lines per minute. The 0770 II printer has several advantages over the earlier 0770-04/05 line printer. Equal throughput is maintained with a smaller footprint, higher reliability, less weight, and lower cost.



Print speeds	950, 1,200, 1,640, or 2,000 lpm depending on character set size
Print line	136 columns
Print density	10 columns per inch
Line spacing	6 or 8 lines per inch (program selectable)
Forms advance rate	Up to 100 inches per second
Data codes	EBCDIC, ASCII, or any code of 6, 7, or 8 bits
Character set	48 standard, up to 384 by feature
Character fonts	OCR-A, OCR-B, H-14, and IBM 407
Forms width	4 to 17 inches (101.6 to 431.8 mm)
Forms length	8 to 22 inches (203.2 to 558.8 mm)
Number of copies	1 original plus 5 carbons

0798 Character Printer

The 0798 printer is a 132-column matrix character printer that can be connected to the paper peripheral controller, a remote printer controller, or to a workstation. The printer is available with a standard 64-character print set, a standard 96-character set, and a 96-character set with lowercase descenders.



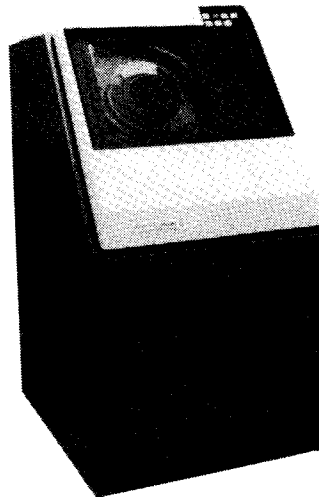
Print speed	37.5 lines per minute, equivalent to 75 lines per minute when printing is bidirectional (200 characters per second)
Print line	132 columns
Print density	10 columns per inch
Line spacing	6 or 8 lines per inch (operator selectable)
Print matrix size	7 x 7 or 7 x 9 by feature
Forms slew rate	6.7 inches per second (17 cm/s)
Forms width	15 inches (38 cm)
Number of copies	1 original plus 5 carbons
Paper feed	tractor control

Magnetic Tape Subsystems

A variety of magnetic tape subsystems can be connected to System 80. The UNISERVO 10/14, 12/16, 20 and 22/24 magnetic tape subsystems are connected through the selector channel. The UNISERVO 22 and streaming magnetic tape subsystems are connected to System 80 through the integrated tape controller. The UNISERVO 10 (type 0871) magnetic tape subsystem is connected by a dedicated controller through the IOP. A maximum of eight UNISERVO 10 tape drives can be connected in this way. The 3200 Series tape subsystem is connected to the selector channel through a SU00212 interface.

UNISERVO 10 Magnetic Tape Subsystem

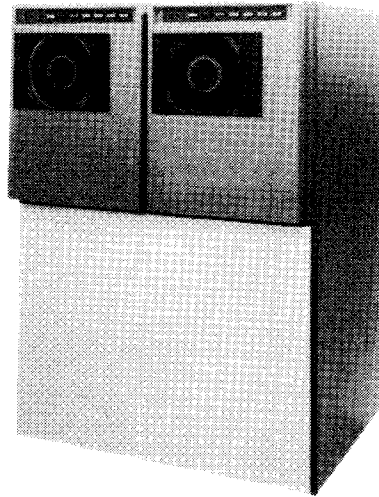
The UNISERVO 10 (type 0871) magnetic tape subsystem is a low-cost unit that has special features, such as automatic tape loading, and uses the ANSI-standard, industry-compatible, wrap-around tape cartridge.



Tape speed	25 inches per second (63.5 cm/s)
Read access time	17 milliseconds
Start/Stop time	17 milliseconds
Reversal time	16 milliseconds
Rewind time	180 seconds for 2,400 feet (731.5 meters) of tape
Data rate	40 KB/s
Recording density	800 and 1,600 bpi in NRZI or PE mode recording 9 tracks; 200, 556, and 800 bpi in NRZI mode recording 7 tracks

UNISERVO 10/14 Magnetic Tape Subsystem

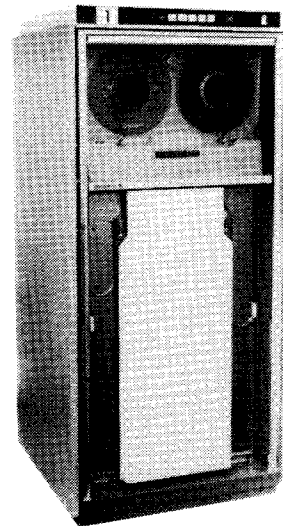
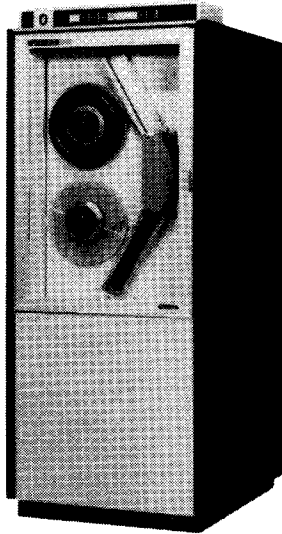
The UNISERVO 10/14 magnetic tape subsystem offers data transfer rates of up to 96K bytes per second. A basic UNISERVO 10/14 subsystem contains two tape drives and a controller in one tape drive cabinet. Up to eight tape drives are supported by one controller and can be included in one subsystem.



	UNISERVO 10	UNISERVO 14
Tape speed	25 ips (63.5 cm/s)	60 ips (152.4 cm/s)
Read access time	17 milliseconds	7 milliseconds
Reversal time	17 milliseconds	6.3 milliseconds
Rewind time	170 seconds	170 seconds
Data transfer rates		
9 track	40 KB/s	96 KB/s
9 track NRZI	20 KB/s	48 KB/s
7 track NRZI at:		
200 bpi	5 KB/s	12 KB/s
556 bpi	13.9 KB/s	33.4 KB/s
800 bpi	20 KB/s	48 KB/s
Recording density	800 and 1,600 bpi in NRZI or PE mode recording 9 tracks; 200, 556, and 800 bpi in NRZI mode recording 7 tracks	

UNISERVO 12/16 Magnetic Tape Subsystem

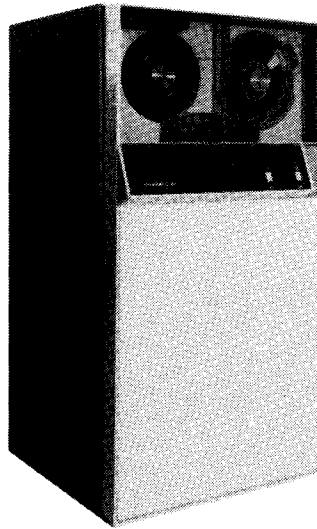
The UNISERVO 12/16 magnetic tape subsystem consists of a controller and up to 16 tape drives. The drives can be all UNISERVO 12 or all UNISERVO 16 drives, or any combination of the two.



	UNISERVO 12	UNISERVO 16
Tape speed	42.7 ips	120 ips
Reversal time	25 milliseconds	10 milliseconds
Rewind time	180 seconds	120 seconds
Data transfer rates		
9-track PE	68 KB/s	192 KB/s
9-track NRZI	34 KB/s	96 KB/s
7-track NRZI at:		
200 bpi	6.4 KB/s	18 KB/s
556 bpi	17.8 KB/s	50 KB/s
800 bpi	25.6 KB/s	72 KB/s

UNISERVO 20 Magnetic Tape Subsystem

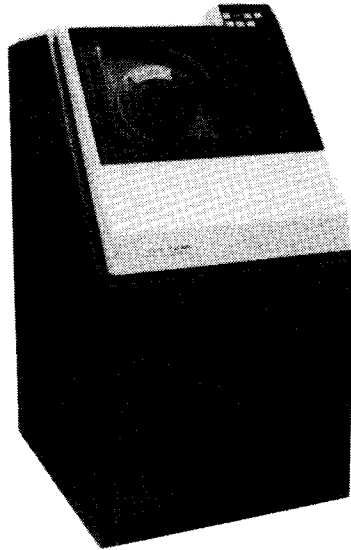
The UNISERVO 20 magnetic tape subsystem is a high-performance tape handling subsystem, consisting of a control unit and from one to sixteen magnetic tape units. The UNISERVO 20 control unit can be configured with a mixture of UNISERVO 12, 16, and 20 tape units.



Tape speed	200 ips
Data transfer rate	320 KB/s
Recording density	1,600 bpi
Reversal time	16 ms
Rewind time	1 minute
Recording mode	9-track PE

UNISERVO 22/24 Magnetic Tape Subsystem

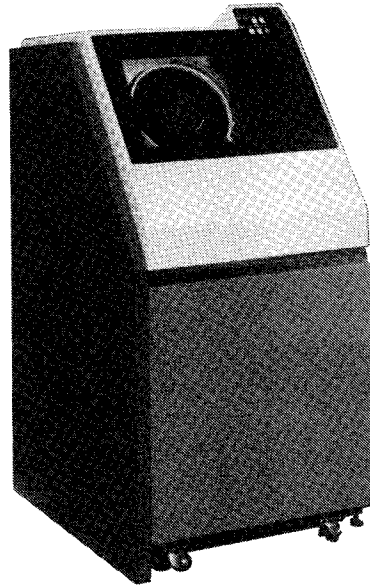
The UNISERVO 22/24 magnetic tape subsystem offers the user a high-speed magnetic tape subsystem in a compact cabinet. Each UNISERVO 22/24 subsystem consists of a controller and one to eight tape drives. UNISERVO 22 and 24 drives can be intermixed in a subsystem.



	UNISERVO 22	UNISERVO 24
Tape speed	75 ips	125 ips
Recording density	800 bpi NRZI 1,600 bpi PE	800 bpi NRZI 1,600 bpi PE
Reversal time	10 ms	10 ms
Rewind time	120 seconds	120 seconds
Transfer rate	NRZI 60 KB/s PE 120 KB/s	NRZI 100 KB/s PE 200 KB/s

UNISERVO 26/28 Magnetic Tape Subsystem

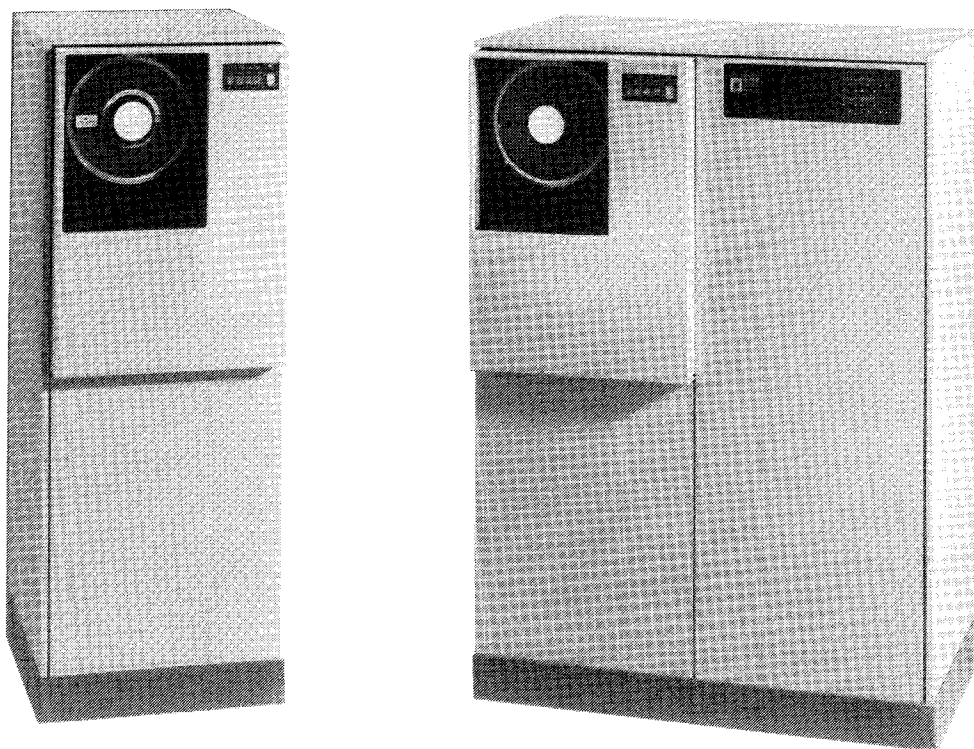
The UNISERVO 26/28 magnetic tape subsystem is a low-cost, high-performance, magnetic tape unit. Each UNISERVO 26/28 subsystem consists of a controller and one to eight tape drives that can be intermixed in a subsystem.



	UNISERVO 26	UNISERVO 28
Tape speed	75 ips	125 ips
Recording density	1,600 bpi PE 6,250 bpi GCR	1,600 bpi PE 6,250 bpi GCR
Reversal time	10 ms	10 ms
Rewind time	120 seconds	120 seconds
Transfer rate	PE 120 KB/s GCR 469 KB/s	PE 200 KB/s GCR 781 KB/s

3200 Series Magnetic Tape Subsystem

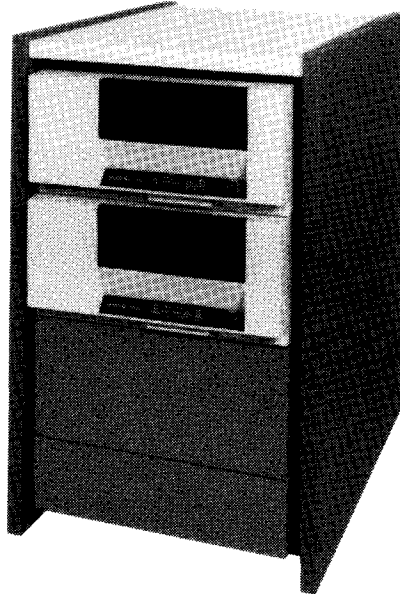
The 3200 Series magnetic tape subsystem is a high-performance magnetic tape unit that ranges from a minimum of one controller/one drive subsystem to a maximum two controllers/eight drives subsystem.



	3244	3266	3288
Tape speed	75 ips	125 ips	200 ips
Recording density	1,600/6,250 bpi	1,600/6,250 bpi	1,600/6,250 bpi
Recording method	PE/GCR	PE/GCR	PE/GCR
Rewind time	55 seconds	55 seconds	66 seconds
Autoload time	12 seconds	12 seconds	12 seconds
Data transfer	120-470 KB/s	200-780 KB/s	320-1,250 KB/s

Streaming Magnetic Tape Subsystem

The streaming magnetic tape subsystem is primarily used as a tape backup for high-capacity disk drives. You can attach a total of eight drives to the integrated tape controller: any combination of streaming tape drives and UNISERVO 22 tape drives, with a maximum of four streaming tape drives.



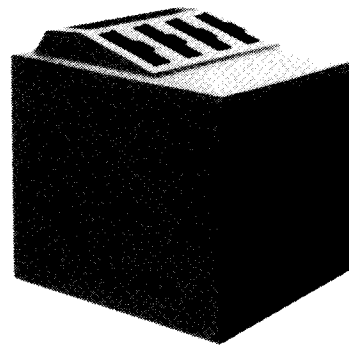
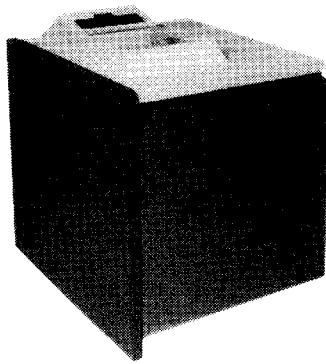
	High Speed	Low Speed
Tape speed	100 ips	25 ips
Interblock gap passing time	6 ms	24 ms
Rewind time	190 seconds	190 seconds
Data rates	160 KB/s	40 KB/s
Data density	1,600 cpi	1,600 cpi
Recording format	9-track PE	9-track PE

Diskette Subsystem

The diskette subsystem supports up to two autoloader drives and two or four manual drives. At least one diskette drive is required in the minimum System 80 configuration. The diskette controller is buffered and performs label checking in data set mode. The controller uses microprocessors and interfaces the IOP through shared direct-access memory.

T8420/8422 Diskette Drives

Both the manual and autoloader drives read one- and two-sided diskettes recorded in either single or double density. You can use diskettes recorded in IBM compatible basic data exchange (BDE) format for data interchange between IBM and Unisys systems. A diskette drive cabinet holds up to one auto-loader and one manual drive, or four manual drives.



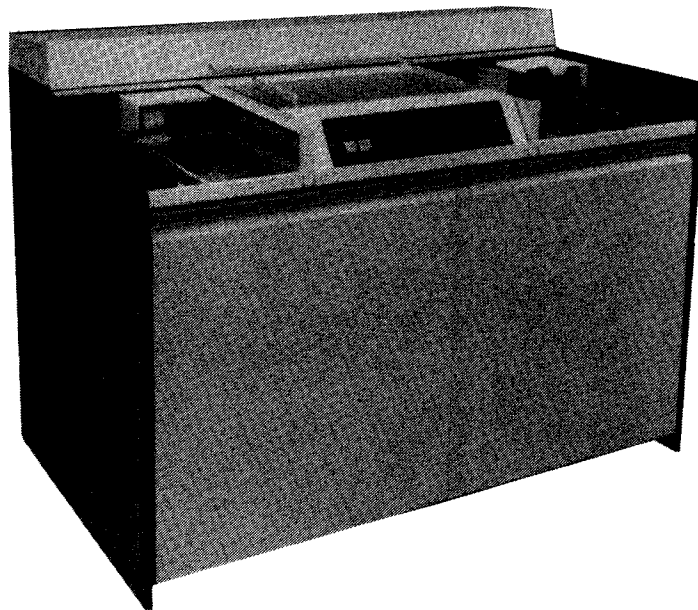
Autoloader hopper/stacker capacity	20 diskettes
Load/unload time per diskette	5 seconds maximum (60 Hz model) 6 seconds maximum (50 Hz model)
Data read/write heads	2/driver
Access time	3 milliseconds track-to-track
Tracks per diskette	77 tracks/side
Tracks per inch	48
Sector size	128, 256, 512, dependent on density method
Data transfer rate	31 KB/s at 256 bytes/sector 62 KB/s at 512 bytes/sector based on sector sequence setup
Bit density	3,400 or 6,800 bpi (8704 or 17,408 bpcm)

Punched Card Subsystems

System 80 offers two card readers and one card punch. One card reader and card punch are integrated with the system and controlled by the paper peripheral controller. The other card reader is freestanding and contains its own controller.

0716 Card Reader Subsystem

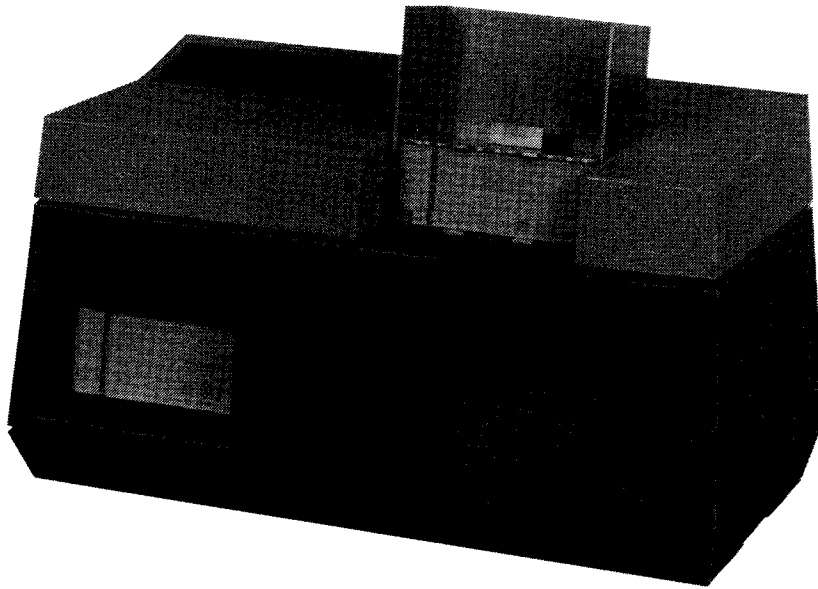
The 0716 card reader subsystem contains its own controller and synchronizer. The synchronizer regulates the flow of data and control signals to and from the reader mechanism.



Reading speed	600 or 1,000 cards per minute
Hopper capacity	2,400 cards
Stacker capacity	2,000 cards each (two stackers)
Read modes	Image mode at 160 characters per card Translate mode at 80 characters per card for EBCDIC, ASCII, and compressed codes
Card sizes	51-column and 66-column short cards, 80-column standard cards
Verification	Multiread validity checking

0719 Card Reader Subsystem

The 0719 card reader subsystem is a tabletop reader that connects to the system through the paper peripheral controller and then to the IOP through shared direct-access memory. No other controller is required in the reader cabinet.



Reading speed	300 cards per minute
Hopper capacity	1,000 cards
Stacker capacity	1,000 cards
Read modes	Image mode at 160 characters per card Translate mode at 80 characters per card
Card size	80-column standard
Read technique	2-column sensors, dual redundant column checking
Read-stating sensing	Column by column

0608 Card Punch/Reader Subsystem

The 0608 card punch/reader subsystem is connected through the paper peripheral controller and then to the IOP through shared direct-access memory. The card punch contains an auxiliary error stacker that holds 100 cards.



Punch speed	75 cards per minute with 80 columns punched or 160 cards per minute with first 28 columns punched; 120 columns per second advance speed
Punch code	Image or compressed
Card size	80-column standard
Checking method	Punch motion check
Feeding method	Column serial, on demand
Hopper capacity	700 cards
Stacker capacity	Primary stacker: 700 cards Auxiliary stacker: 100 cards
Read/validity check	Reads prepunched cards at 160 cards per minute

Communications Subsystems

The single line communications adapters (SLCAs) provide system communications capability. In addition, you can get high-speed communications capabilities through the direct connection of a DCP/15, DCP/30, DCP/40, or DCP/50 through the selector channel. All communications services of the distributed communications processors are available to the users of these devices.

The multiple line communications multiplexer component of the IOP supports up to 14 half-duplex SLCAs operating at medium or low speeds, or up to 7 full- or 12 half-duplex SLCAs operating at high speed.

Remote workstations, terminals, and other communications devices are supported through the SLCAs. Various communications lines can be used with the SLCAs:

- Digital data network for distributed data processing
- Public data networks
- Private leased lines
- Automatic answering/autodial capability
- Telegraph
- Military communications networks

The SLCA controls communications line characteristics and the communications device attached to the line. Each line can accept devices operating at speeds of up to 56K bits per second. SLCAs also perform special character recognition, integrity checks, data transfer control between main storage and devices, and other required control functions.

The table of communications functions that follows lists the basic protocols used with the SLCA.

UTS 20

The Universal Terminal System 20 (UTS 20) is a versatile, single-station display terminal used to send, receive, display, and manipulate data. As an efficient I/O and display device interfaced with your host system, this terminal provides the basic capabilities required at remote locations – online interactive data transfers and a hard-copy printer.

The functions of this nonprogrammable terminal are stored in an electronic program cartridge, which you can easily install in the back of the terminal.

Communications Functions

Protocol/Function	Line Speed	Mode	Interface
UTS 20, UTS 30, UTS 40, UTS 400, PC, UNIX	2,400 - 19.2K bps	Half-duplex, synchronous	RS-232-C or MIL-STD-188
BSC	2,400 - 9,600 bps	Half-duplex, synchronous	RS-232-C or MIL-STD-188
UTS 10/TTY (or equivalent)	50 - 9,600 bps	Half-duplex, asynchronous	RS-232-C or MIL-STD-188
Universal Data Link Control (UDLC)	9,600 - 56K bps	Half-duplex or full- duplex synchronous	RS-232-C/X.21. BIS or CCITT V.35
NTR	4,800 - 56K bps	Full-duplex, synchronous	RS-232-C or MIL-STD-188 or CCITT V.35
SVT 1120/1123/1124 UTS 4020/4040	2,400 - 19.2K bps	Half-duplex, synchronous	RS-232-C or CCITT V.24

UTS 30

The Universal Terminal System 30 (UTS 30) single station is a low-cost, intelligent display terminal specially tailored in three different versions for sending, receiving, manipulating, and displaying data.

- **Basic UTS 30** - The powerful UTS 30 capabilities are loaded from diskette by a Unisys 8439 double-sided diskette subsystem or provided by an easily installed program cartridge. This version can be used in a communications network with other Unisys terminals.
- **Programmable UTS 30** - This efficient display terminal provides versatile capabilities for use at remote locations. The operating modes give you the advantages of hard-copy printers, peripheral data storage, online interactive data transfers, offline data handling potential, user programmability, and host-processor-initiated retrieval of data in peripheral storage.
- **Twisted pair interface** - The UTS 30T version can be downloaded without a program cartridge or system diskette.

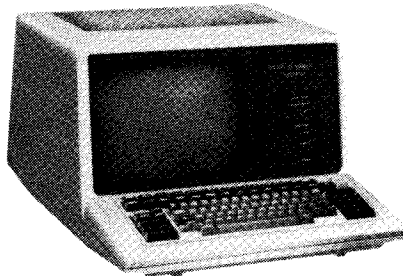
UTS 40

The Universal Terminal System 40 (UTS 40) single station is an intelligent, programmable display terminal used to send, receive, and display data. The UTS 40 is an efficient and versatile display terminal that can be interfaced with the host system. The UTS 40 offers many useful capabilities for remote locations: hard-copy printers, peripheral data storage, online interactive data transfers, offline data handling potential, programmability for user-selected functions, and host-processor-initiated retrieval of data in peripheral storage.

UTS 400

The Universal Terminal System 400 (UTS 400) is a powerful, programmable terminal display device. A variety of presentation formats provide a total display capacity of 960, 1,024, 1,536, or 1,920 characters. The UTS 400 can operate as a data entry device or as a display device conveniently located at the central computer site or at a remote location.

You can configure the UTS 400 as a master terminal with up to two slave terminals, or as a controller with up to six slave terminals. You can also use it with UNISCOPE 100 or UNISCOPE 200 display terminals in multidrop or multiplexer configurations.

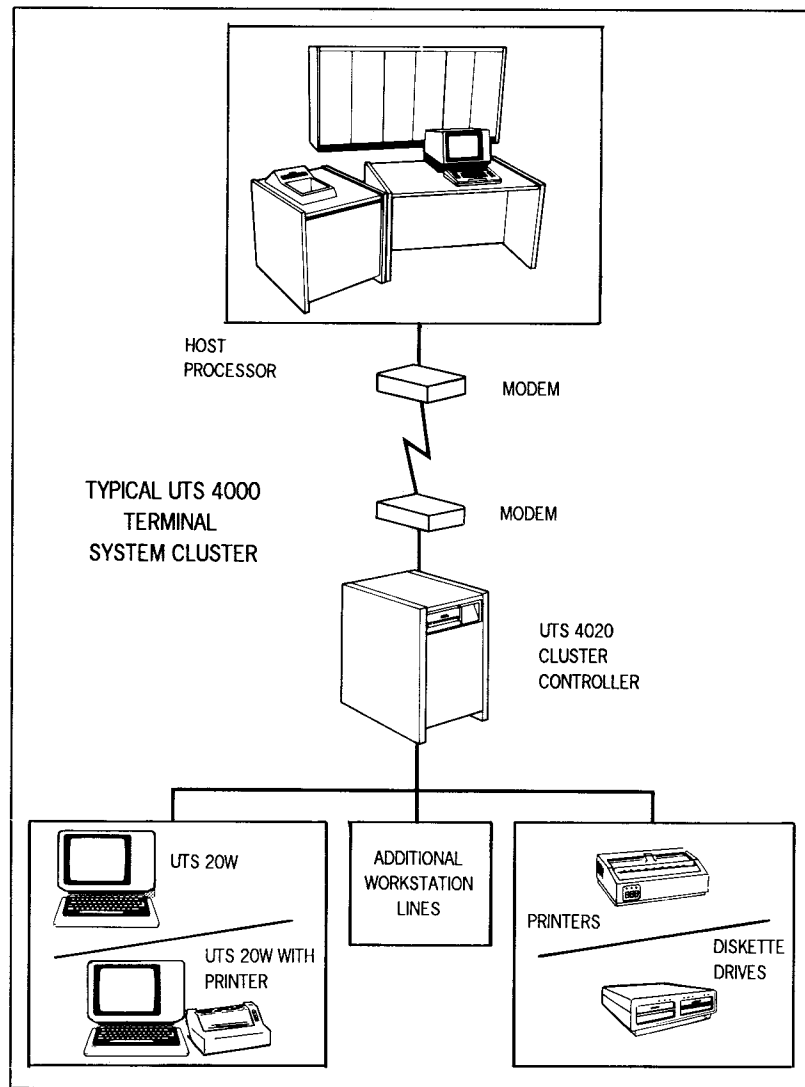


UTS 4000

The Universal Terminal System 4000 (UTS 4000) is a comprehensive family of communications devices with a wide range of interactive communications capabilities. You can use the UTS 4000 for a variety of applications, from simple data entry and retrieval to sophisticated distributed data processing. Using interchangeable components, you can configure the system to meet the needs of any communications user.

The UTS 4000 uses a terminal cluster architecture with a programmable controller unit attached to the host processor. Terminals and other peripheral devices are connected through the controller. Available terminal types are video display devices and include editing and programmable terminals. Peripheral devices include a desktop diskette unit supporting 1-megabyte diskettes and a variety of terminal printers.

The UTS 4000 control unit (the UTS 4020) is a programmable, microprocessor-based unit capable of acting as a stand-alone processor, as well as a communications controller. Provided with an integrated diskette unit, random-access storage, and a complement of software elements, it is one of the most advanced communications devices available. The stand-alone processing capabilities of the UTS 4020, coupled with its advanced communications features, make the UTS 4000 ideally suited for inclusion in a DDP network.



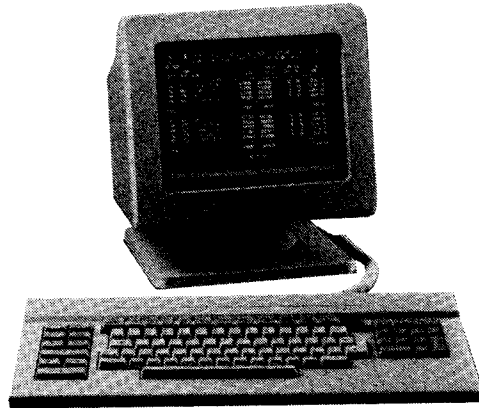
SVT 1120 Terminal

The SVT 1120 terminal is UNISCOPE control-code-compatible and functions similarly to the 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 shutoff and a display intensity control.

Keyboard-selectable languages supported include: domestic 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 Remote Workstation

The SVT 1123 remote workstation 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 Remote Workstation

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

DCP 15/30/40/50

The DCP 15/30/40/50 can be connected to the System 80 through a communications subsystem. In addition, a connection via a dedicated selector channel can also be provided.

Additional Communications Devices

In addition to the Unisys devices listed, the following devices can also be attached as communications terminals:

- Unisys PCs
- MAPPER 5
- UNIX systems
- IBM 3270 terminal system
- Teletype models 33, 35, 37, and 38

While these devices can be attached as communications terminals, all services available to the normal communications user cannot be available to users of these devices.

Hardware Summary

The following table summarizes the system hardware characteristics of System 80 models 10-20:

System Characteristics

Characteristics	
System orientation	Disk, with concurrent interactive and batch processing
Central processing unit	Two sets of 16 general-purpose registers Four floating-point registers Priority controller for access to any two units on a priority basis Four-stage pipeline Maximum system bus transfer rate of 16 MB/s
Main storage processor	64 relocation registers for address conversion (48 for user programs, 16 for operating system use) 48 protection keys, error correction, storage protection
Main storage unit	2 MB expandable to 8 MB (model 10), 4 MB expandable to 12 MB (model 15), 4 MB expandable to 16 MB (model 20). Expansion in 2 MB (model 10 only) and 4 MB increments.
Instruction cache	16 KB capacity, 200 ns access (model 20 only)
Operand cache	16 KB capacity, 200 ns access (model 20 only)
Floating-point processor	Floating-point and multiply accelerator (model 20 only)
Selector channel	Maximum of six selector channels supporting the following: * Up to three integrated disk control units and four freestanding control units, or eight freestanding control units. * Supports removable disk drives with capacities of 29, 58, 72, 100, and 200 MB. Fixed media disk units with capacities of 118.2, 491, 776, 1,964, and 2,464 MB.

continued

System Characteristics

Characteristics	
Selector channel (continued)	<ul style="list-style-type: none"> * Additional peripheral controllers, supporting the following peripherals: <ul style="list-style-type: none"> - 0716 card reader - UNISERVO 10/14, 12/16, 20, 22/24, 26/28, and BT3244/66/88 tape drives - 0770 freestanding line printer (2,000 lpm) - 0776 printer - 9246-14B freestanding line printer (1,200 lpm) * Connected to a DCP 15/30/40/50 and dedicated to a channel (no other DCPs or devices connected).
Input/output processor	<p>Maximum of two IOPs per system, each supporting up to eight device controllers consisting of:</p> <ul style="list-style-type: none"> * Diskette controller with up to four manual drives, or two autoload plus two manual load drives. No more than a total of four diskette drives per system. * Workstation controller supports eight workstations with a maximum of 120 workstations per system. * Paper peripheral controller supporting: <ul style="list-style-type: none"> - Two type 0719 card readers, or one card reader and one type 0608 card punch. - Two local printers, from 200 cps to 1,200 lpm not to exceed 1,500 lpm. * Maximum of 8 paper peripheral controllers per system * Remote printer controllers, each supporting one type 0789 or 0798 printer

continued

System Characteristics

Characteristics	
Input/output processor (continued)	<ul style="list-style-type: none">* Integrated tape controller capable of supporting a total of eight magnetic tape drives in the following combinations:<ul style="list-style-type: none">- Up to four streaming tape drives- Up to eight UNISERVO 22 tape drives- Any combination of the two, not exceeding four streaming tape drives or eight tape drives total.* Dedicated tape controller channel, supporting up to eight UNISERVO 10 (type 0871) tape drives.* Multiline communications multiplexer interfaces up to 14 low- or medium-speed SLCAs or up to 7 full- or 12 half-duplex, high-speed SLCAs.

NOTES



NOTES



NOTES

