SYSTEM OVERVIEW

IBM

3) System Modularity

IBM's System/38 is a general purpose data processing system designed to significantly improve the user's productivity in developing, maintaining and enhancing applications in an interactive workstation environment.

The system provides 512K to 1,536K positions of main storage, 64.5 to more than 2672.6 million bytes of nonremovable auxiliary (disks) storage, a diskette magazine facility, and a system console with keyboard and display. A multi-function 96-columm card reader/punch, up to two 650 line per minute system printers and up to four 3410 magnetic tape subsystems may be attached. The system supports direct attachment of up to 40 local 5250 workstation displays and printers and a large number of remotely attached workstations through SDLC teleprocessing lines.

The workstations can be placed where they are needed in the organization (in the offices, departments, plants and the warehouses) so that company personnel can share a common data base and the processing power of a computer. System/38 supports this workstation environment with the function required to maintain up-to-date business data available for retrieval on request by any of the workstation users. By providing these users with information that is current and accurate, the efficiency of an organization can be greatly improved.

The workstation environment requires many unique functions such as the ability for workstation users to share programs, data files, and system resources without significant delays to any user. This environment also requires the ability to access data records by more than one access path (index). For example, one user may need to access inventory item records by item stock number while another user is accessing the same records by vendor number.

On most current systems the unique functions required by the workstation environment have either been added on to the existing batch programming system or they are programmed by the user in the application programs. In contrast, System/38 is an entirely new system developed specifically for the workstation environment. The needed function has been designed into the system. Major advantages in the level of

system function and ease of use of that function enable the System/38 user to develop workstation applications in a shorter time and with a lower level of expertise than was required on previous systems.

System/38 can support a range of environments from one consisting almost entirely of batch processing to one that makes extensive use of workstation applications. System/38 is designed to allow the user to make easy transitions through this range of environments while making optimum use of system resources. In general, installations that make full use of workstations still require the execution of a considerable amount of batch work.

Therefore, the System/38 has been designed to manage a mixture of batch and interactive work providing fast response to workstation user requests and suitable throughput for batch jobs.

Upgrading the System/38 to provide increased capacity is a relatively simple chore. When adding memory or auxiliary storage devices, the system will automatically start using these facilities when the configuration record is changed. User programs are not affected by these changes because of the System/38's use of single level storage addressing. This technique provides for addressing up to 281 trillion bytes.

The System/38 will automatically start using additional capacities when the configuration record is changed. Providing a minimum impact on programs and the user when the system is changed to meet the increasing demands of the city.

SYSTEM/38 HARDWARE

IBM

SYSTEM/38 FEATURES

7A) Overview

In developing System/38, first the desired functions were specified. It was then determined which of these functions should be performed by the control program and which part should be performed by the machine. Next, a machine that consists of electronic circuitry and micro-programs was designed with a functionally high level interface. The machine does much more than just add, subtract, move data and execute I/O. The machine performs complete functions such as: Finding a file or program in system storage, locating a record in a file through a key (index), initiating a new process, manipulating queues and allocating storage.

The high level System/38 machine interface ensures compatibility of the control program facility and the users' programs implemented on different models of the machine. New physical devices can be added to the system without requiring reprogramming. A user can expect greater reliability because basic system functions are built into the machine.

Additional high usage micro-programmed functions can be implemented in the circuitry of the machine to provide improved performance without impacting the control program facility or the users' program. New functions can be added to the high level machine interface in an upward compatible manner because of its extendable design.

IBM's emphasis on advanced technology and the development of a highly integrated system enables System/38 to provide users with an improved level of productivity, ease of use, data integrity, functional capability, reliability and non-disrupted growth in applications.

HARDWARE SUMMARY

System/38 is available with a variety of input/output devices and a wide range of storage capabilities.

7A) IBM 5381 SYSTEMS UNIT

By design, System/38 is oriented toward a multi-user system environment. Logical functions within the system unit are implemented in an advanced high density hardware technology.

The major functional units of the system consists of the following:

- Processing unit and its related control storage
- Virtual address translator
- Input/Output channel
- Main Storage

The primary characteristic of the System/38 identifying it as a major advance in computer system architecture is its high level machine interface. This interface embodies many of the basic supervisor and resource management functions previously found in system programs.

The high level operations performed by the machine interface instruction offer the user the logical functions desired without dependence on the machine implementation. The power of these instructions is illustrated by data base operations that retrieve, update and sort data records.

The access path to information is machine controlled. This permits effective enforcement and automatic serialization of concurrent operations on the same information. Pointers used to address information are tamper-proof to prevent obtaining addressability to unauthorized information or virtual storage used by the micro-code. These features provide greater data integrity and security than available on previous systems.

Programs are translated before execution into microcode to achieve greater efficiency. Program variable attributes allow the machine to automatically perform data type conversions and allocate program work storage for these variables.

overhead for translation effort?

Input/output operations offer greater device independence through the use of a machine device support component and System Network Architecture. The intricacies of the channel communication networks and asynchronous device operations are handled by the machine.

System/38 incorporates all of these features and more into the machine hardware and microcode. This high level of function is standard on every machine model regardless of storage size, processor type or device configuration.

7A) Hardware Organization of the System/38

System/38 manipulates a unit of execution called the "task". All computer systems need to control execution and, in multiprogrammed systems like System/38, switch between units of execution, i.e., tasks. Traditionally, an interrupt structure with a fixed number of interrupt levels or classes, built on the hardware, is transformed by a software supervisor into a multilevel, interrupt-driven system to bridge the gap between the actual hardware and the abstract concepts of multiprogramming. The System/38 replaces this interrupt structure with a single tasking mechanism which is used to control all processing.

A multilevel, queue-driven task control structure is implemented in microcode and hardware on the System/38. A task dispatcher implemented in microcode allocates processor resources to prioritized tasks. I/O and program processing tasks are integrated in a common dispatching structure, with their priorities adjusted for system balance. I/O processing takes place when system resources are available, not when an I/O interrupt occurs.

I/O and program processing requests are stacked in main storage on a linked list called the task dispatching queue (TDQ). The task dispatcher selects the highest priority request from the TDQ and gives it control of the processor. Instructions associated with this task, known as the active-task, are executed until control is passed to another task.

A set of system control operations (SEND and RECEIVE) are used to communicate between tasks and to pass control between tasks via the task dispatcher.

If the active task is to communicate with another task, it does so by sending a message to a queue in main storage known to both tasks. If the active task is to obtain a message from a queue, it executes a RECEIVE operation. If the message is available on the queue, the message is passed to the active task and processing continues. If the message is not available (e.g., it has not yet been sent), the active task is made inactive and the task waits for the message. The task dispatcher is then invoked to select the new active task from the TDQ. The task dispatcher is also invoked on a SEND operation if a task of higher priority than the active task is waiting for the message. If the waiting task is of lower priority than the active task, the task dispatcher is not invoked, but the processing request for the waiting task is placed on the TDQ.

I/O System/38 is implemented with a queue-driven command structure using the SEND/RECEIVE mechanism to pass information across the I/O interface. To a task, a device looks like another task. Commands to devices and responses from devices are exchanged in the same way that messages are communicated between any two tasks in the system. The messages sent to the devices are specially formatted and contain the device commands. In addition to individual commands, a complete channel program can be sent as a single message. Because a queue structure is used, command stacking is automatic. In a similar manner, the device sends response and status information back to a task via a main storage queue. Note that only commands and responses use the queueing structure; data transfers between devices and main storage are direct.

High-level call/return functions are directly supported by another set of system control operations which provide the linkage mechanism between routines executing within the same task. The performance of programs written using structured programming techniques is enhanced by the use of this mechanism. The same linkage mechanism is used by the hardware to report program exceptions. With this mechanism, exceptions for any task (including such things as page faults) execute at the same priority level as the task itself. A low priority task incurring an exception will not interfere with the excecution of higher priority tasks.

Summary

The hardware implementation of System/38 provides the foundation on which the high-level machine architecture is built. Through the use of advanced LSI technologies, System/38 achieves a high level of processor performance and reliability. The use of intelligent controllers for I/O device attachments distributes the I/O workload throughout the system.

A unique aspect of the System/38 hardware and microcode is the incorporation of very powerful control functions. These functions provide a single mechanism which is used to control all processing in the system. Other high-level functions implemented in the microcode further enhance the flexibility and performance of the system.

7A) Processing Unit

The processing unit is the control center of System/38 and supplies the facilities for:

- Fetching and storing data
- Arithmetic and logical processing of data
- Executing instructions in a particular sequence
- Controlling communication between main storage and the input/ output devices

In performing its required functions, the processing unit uses internal storage called control storage. The purpose of control storage is to provide random access storage with a powerful, high level 32 bit word, micro instruction set (microcode) used by the processing unit. Many of the functions normally coded in system supervisors are implemented in the System/38 microcode resulting in a more efficient system.

7A) Main Storage Capacity

Main storage capacity provides the system with directly addressable storage for data and programs. The basic main storage module consisting of 512K (524,288) bytes of storage is equipped with error-correction hardware to automatically detect all single and double bit errors and to automatically correct all single bit errors. Three main storage capacities are available with either processor:

- 512K (524,288) bytes
- 768K (786,432) bytes
- 1024K (1,048,576) bytes

In addition, these units will be available with the 5xx model:

- 1280K (1,310,720) bytes
- 1536K (1,572,864) bytes

The main storage interface between the input/output channel and the processing unit is 4 bytes wide. Main storage performs at a nominal cycle time of 1,100 nanoseconds for Model 3xx, while main storage in the Model 5xx functions at 600 nanoseconds.

7A) Time of Day Clock

The time of day clock is a timer that runs continuously whenever system power is on. The clock is not affected by system status (stopped or running). Functions are supplied to the user to set the clock and to schedule work execution, such as perform an action after the expiration of a specific time interval.

7A) Automatic Initial Microprogram Load

If commercial power is interrupted, this optional feature automatically initiates a power-on sequence after commercial power is restored. Once the control program facility has been loaded into the system, this feature causes a branch to a user-provided recovery program to resume operation.

7A) Power Key Lock

This optional feature supplies a key-controlled switch in series with the power on key to restrict the power-on cycle to employees who have the key.

7A) Storage Management

Main storage is not divided into partitions. Therefore, most operating and design considerations (i.e., program size or which programs can execute concurrently) are eliminated.

System/38 storage management offers advantages such as:

- The system's storage management function maintains a directory of locations for all objects (such as files and programs) in the system. The programmer no longer has to consider specific file location, disk volume, drive, or space requirements in the design of his program.

- The system allocates storage space as it is required. Therefore, if more space is needed for a file during the execution of a program, the system obtains the space without interrupting the program. Also, as more storage capability is added to the system that space is automatically used by the system without requiring change to executing programs or control language commands.

- Treating all of auxiliary storage as a part of one continuous volume of space permits any file to exist on multiple disk enclosures. A file is never limited in size by the amount of available space on a single disk enclosure.

7A) Virtual Address Translator

The Virtual Address Translator, as used by storage management, permits users to program without regard to the actual amount of main storage required for execution. In addition, the virtual address translator assures better storage utilization, increased capability to use multi-tasking and rapid execution of programs. The virtual address translator translates real storage addresses to and from virtual storage addresses.

7A) Internal Input/Output Channel

The high speed internal input/output channel, with an aggregate transfer rate of over 2 megabytes per second, connects the Processing Unit, Main Storage and the input/output devices. The channel permits the overlapping of input/output operations with processing.

7F) System Console Keyboard/Display

System/38 is designed to operate in an unattended environment. The System/38 console includes a keyboard and a display screen. The console is used primarily for interactive communications between the system operator and programs (Control Program Facility and user programs) executing in the system and between the system operator and workstation users.

The system console display screen can display up to 16 lines of information having 64 characters per line. The characters on the console display screen are larger than those displayed on the workstation. This improves readability for the operator who may be some distance away. The intensity of the display can be changed by the control on the operator service panel. In addition to the standard attention indicator, an optional feature is available that provides a back lighted indicator, an audible alarm and a volume control. These features alert the operator to pending messages that require operator attention.

The system console keyboard is used by the system operator when entering Control Language commands and when responding to MENU's, prompts and messages for operator action.

The keyboard contains:

- Standard alphameric keys for entering alphabetic (upper and lower case) and numeric data
- Cursor movement keys to position the cursor on the display screen
- Twenty-four (24) easily identified command function keys for requesting specific program functions.

The operator/service panel has lights, switches and keys that are used for communication between the system operator and the system.

- Lights indicate conditions in the processing unit
- The major switches are used to power on the system and initiate the microprogram load process. Additional switches may be used by the Service Representative to control the operation of the system

7F) System Control Adaptor

The system control adaptor, although not visible to the user, is the interface between the system and the operator/ service panel. Whenever the system is initially powered on, the system control panel provides an automatic diagnostic check to the major units of the system unit (such as auxiliary storage, diskette magazine drive and the system console).

7B) Auxiliary Storage

On-line auxiliary storage is supplied by the data storage function. The data storage function provides the System/38 user with auxiliary storage (magnetic disks) starting with a minimum of 64.5 megabytes and increasing up to more than 2.6 billion bytes. This lets each user select the amount of storage necessary to contain all data on-line making the data readily available for processing.

Disk enclosures 1 and 2 are packaged in the base enclosure unit. Disk enclosures 3, 4, 5 and 6 are in the expansion enclosure.

The performance and physical characteristics of data storage functions are:

- Access times
  - 9.6 milliseconds average rotational delay
  - 27 milliseconds average seek time
  - 9 milliseconds minimum seek time
  - 46 milliseconds maximum seek time
- Data transfer rate - up to 1,031,000 bytes per second
- Number of sectors: each track has 33 sectors, 0 through 32. One sector, normally unassigned, is for alternate use. That is, if a sector in a track fails, its use is assigned to the same track in the alternate sector.
- Size of sector: each sector can accommodate 512 bytes of data.
- Capacity: 64,520,192 bytes

7B) 3370 Direct Access Storage Device

The 3370 provides an advanced, fixed media, Direct Access Storage with:

- The ability to have active data directly online to support data base data communications and other applications.
- Fixed, sealed Head Disk Assemblies (HDAs) which provide large capacity, high density data storage, eliminate operator handling and reduce exposure to outside contamination.
- Fixed Block Architecture which provides a linear contiguous data addressing scheme for ease-of-use and is part of System/38 Single Level Storage Management.
- Capability of correcting single data error bursts of up to nine bits span as well as detecting all single error bursts of up to sixteen bits span.
- Up to four (4) 3370 units, 2285.5 MB total, can be attached to any 5381 Model 5XX to provide total auxiliary storage up to 2672.6 MB. When used in conjunction with the 64.5 megabyte disks.

Two models of the 3370 Direct Access Storage device are available for attachment to the System/38 5381 System Unit (Model 5 only). The 3370 Model A11 contains the control adapter functions required for attachment. The 3370 Model B11 attaches through an A11 unit. Up to three 3370 Model B11s can be attached to a 3370 Model A11 for a maximum of four units.

Each 3370 disk unit has a single spindle of disks which are accessed by two independent, movable access mechanisms (actuators), each accessing one-half the data. Seeking with either actuator may be overlapped with seeking and/or reading/writing of the other actuator.

The 3370 Head Disk Assembly (HDA) consists of head and disk components assembled as units. These units are field replaceable (removable) only by Customer Engineering. An HDA contains two actuators, each separately addressable.

Each actuator spans 558,000 blocks or 285,696,000 bytes. Each HDA contains 571,393,000 bytes of usable data.

The data rate is 1.859 megabytes per second (nominal).

Access Time:

Minimum Seek	5 ms
Average Seek	20 ms
Maximum Seek	40 ms
Average Rotational Delay	10.1 ms

7F) Diskette Magazine Drive

The diskette magazine drive, standard on all System/38 models, is used primarily for the following functions:

- Save/Restore - The selective backing up of data, programs, or files kept on-line in auxiliary storage. It also permits easy reloading of data files, if required, for recovery.
- Diskette Input/Output - The interchange of data where diskettes are exchanged with other systems and devices, such as System/32, System/34, IBM 5110 computer and IBM 3741 Data Station.
- System Servicing - The loading of system diagnostic programs, engineering changes, program changes and new program releases from the IBM Program Information Department (PID).

The diskette magazine facility is a sequential save/restore device that reads and writes data, at high speeds, from and to a small, flexible magnetic disk called a diskette. Each

diskette is permanently enclosed in a protective jacket and is removable from the diskette drive. The diskette magazine drive is two magazine positions and three individual diskette slots. Each magazine can contain up to ten diskettes resulting in a total capacity of 23 diskettes.

The individual diskette slots are generally used for small quantities of data (for example, when entering data from individual diskettes received from 3741 data station operators). For large quantities of data, operators can load their diskettes into a magazine. The system operator can then load up to 10 diskettes in one physical operation. Typically, processing begins with the first diskette in magazine number one and can automaticaly proceed diskette by diskette through both magazines. If magazine number one has been replaced the processing can continue again with this magazine. The maximum capacity of the diskette storage facility containing two full magazines is approximately 24 megabytes. Advantages of the diskette magazine facility include:

- Less operator intervention - up to 24 megabytes of data may be written on two magazines before system operator action is required.
- Improved operation convenience - with the magazine concept, diskette handling, labeling, and storage are simplified.
- Random access to any data set on any diskette
- Multi-function device operation
- Save/restore
- Spooling
- Copy
- RPG III file operations

7C) Communications Adapters

System/38 provides the capability to communicate with multiple workstations at local and remote locations. Workstations can operate concurrently with no apparent interference (depending upon the system workload) in the operation of any other application or individual workstation.

Two optional attachment methods exist: one is through the workstation controller, the other through the communication attachment.

7C2) Workstation Controllers

The workstation controller provides for the local attachment of IBM 5250 Information Display System devices. These devices can be connected directly to System/38 through this low cost attachment.

There is no need for common carrier provided service to attach the workstations. The limitations of telecommunication data rates do not pertain to the workstation controller.

7C2) Workstation Controller (Basic)

This workstation controller is standard on all models of System/38. The controller provides eight ports for attaching workstations directly to System/38. These eight ports permit the attachment (in a single and multi-point arrangement) of up to 12 workstations.

Twinaxial cable is recommended for attaching the workstations because it allows a maximum length of 1,525 meters (5,000 feet) and multiple workstations (7) to be attached to a single cable. This cabling arrangement requires use of the 5250 Cable Through feature.

Those users who now have IBM 3270 Information Display System devices can use the existing one wire (coaxial) shielded cable when installing IBM 5250 Information Display System devices. A twinax/coax adapter is required at both ends of the cable to connect the 5250 devices.

The maximum allowable length of a coaxial cable is 610 meters (2,000 feet) and only one workstation may be attached to a coaxial cable. A second workstation may be attached through the Cable Through feature. Twinaxial cable must be used between the first and second workstation, and the maximum allowable distance between these two workstations is 30 meters (100 feet).

7C2) Device Control Expansion

This special feature permits the workstation controller to support up to eight additional workstations. This feature does not provide any additional ports but provides necessary control storage to support eight additional workstations attached through the ports supplied with the workstation controller.

Note: This feature cannot be installed in conjunction with the device interface expansion feature.

7C2) Device Interface Expansion

This special feature provides the necessary control and eight more ports for the attachment of additional workstations.

Note: This feature cannot be installed in conjunction with the device control expansion feature.

7C2) Workstation Controller (Second)

This workstation controller is a special feature on all models of System/38. This controller provides support identical to that of the Workstation Controller (Basic), that is 20 additional locally attached devices.

7C2) Communication Attachment
8F)

The communication attachment provides the necessary system control for the direct attachment of up to 4 remote communications lines. The communication attachment permits the attachment of those devices that require data transmission service (such as private or common carrier supplied lines).

Voice grade transmission service at 600, 1,200, 2,400, 4,800, 7,200 and 9,600 bits per second (bps) across private or common carrier lines is supported.

Support is supplied for current analog type data communications facilities, and also for the Dataphone* Digital Service Network from the American Telephone and Telegraph Company (U.S. only). Analog type facilities are given support either through the integrated (under the systems covers) 600/1,200 bps modem or through the external modem attached through the EIA/CCITT interface features. Digital facilities using no modems are given support at 2,400, 4,800 or 9,600 bps through the Dataphone Digital Service Adapter.

Through this communications support, System/38 can communicate as a terminal system to IBM System/370, CICS and IMS using Synchronous Data Link Control (SDLC) as a part of the System Network Architecture (SNA).

The System/38 Control Program Facility provides communications support for System/38 as a terminal system to a System/370, Model 115 through 168, operating under DOS/VS, OS/VS1 and OS/VS2 (or any of three operating systems when running VM/370).

* Registered Trademark of American Telephone and Telegraph Company

The communications access method is VTAM or TCAM. The communications controller is a 3704 or 3705 operating under Network Control Programs/VS (NCP/VS). The System/38 is defined to the System/370 as a LU Type 1 3770.

7C2) 5250 INFORMATION DISPLAY SYSTEM

The 5250 Information Display System which consists of table top display stations and printers is designed for data entry and inquiry applications. These workstations can be set up in areas away from System/38, such as the sales counter, order entry department and shipping department.

The 5250 Information Display System offers:

- o A 5251 Model 2 and 12 Display Station that provide remote communications with System/38 using SDLC line discipline;
- o A 5251 Model 1 and 11 Display Station, 5252 Dual Display Station and 5256 Printer that can be cable connected directly to System/38;
- o A Cable Through feature for connecting 5251 Model 1 and 11 Display Stations, 5252 Dual Display Stations and 5256 Printers in series;
- o A Cluster feature that allows the 5251 Model 2 and 12 Display Station to control up to eight 5251 Model 1 and 11 Display Stations, 5252 Dual Display Stations or 5256 Printers, or a combination of all three.
- o The functional capabilities to:
  - Display and print upper case and lower case alphabetic characters.
  - Display highlighting functions such as blink, column separator, high intensity, non-display, reverse image and underscore.

7C2) IBM 5251 Display Station

The 5251 Display Station has these characteristics that are common to all models:

- o A display screen (two sizes are available)
  - Models 1 and 2 - 960 characters (12 lines of 80 characters)
  - Models 11 and 12 - 1,920 characters (24 lines of 80 characters)
- o A movable keyboard with several key arrangements
- o A keyboard COMMAND key for selecting up to 24 additional programmable command functions
- o (Models 2 and 12 only) An internal communications adapter for the remote attachment of a common carrier communications line using the SDLC line discipline

7C2) 5252 Dual Display Station

The 5252 Dual Display Station is designed for use by two operators in high production data entry application or where multiple display stations are required at a specific location.

The Dual Display Station has two movable keyboards that are connected by a short cable to a common display unit. This display unit is divided in half logically and optically into two independent workstations which allow each operator to view a separate 960 character display with up to 12 lines of 80 characters each. The 5252 provides the same display control on highlighting of individual fields as a 5251 and also uses the same keyboard.

7C2) 5256 Printer

The 5256 Printer has these characteristics:

- o A printing speed of 40, 80 or 120 characters per second
- o Bi-directional, serial matrix printing
- o Full look-ahead capability for most efficient throughput
- o A maximum print line of 132 characters
- o The capability to use individual or continuous forms

- o Print position spacing of 10 characters per inch
- o Selectable line spacing of 6 or 8 lines per inch
- o The capability to print on continuous forms that have an original and five carbons
- o An audible alarm feature
- o Forms control to control spacing and skipping of forms that have sprocket feed holes

7E) 5211 PRINTER AND 3262 PRINTER

The System/38 supports one model of the 5211 Printer and two models of the 3262 Printer as the systems printers.

Type	Model	Rated Speed (lpm)
5211	2	300
3262	A1, B1	650

Note: The A Model directly attaches to the System unit.

Both types of printers have improved set up features and quiet operation. The quiet operation is achieved by feeding paper completely within the covers. Operator set-up time is minimized because forms control, as well as 6 or 8 line per inch selection, is programmable.

Both the 5211 and the 3262 Printers are line printers; that is, they print one complete line at a time, up to 132 characters per line. Speeds of the 5211 and the 3262 Printers will vary depending on character set size, the number of characters and form skips associated with the output to be printed. The printers can print an original copy with up to 5 carbon copies on paper from 3.5 inches to 15.25 inches wide (16 inches wide on 3262) and from 3 inches to 14 inches long. The outward physical appearance of the 3262 Printer is identical to the 5211 Printer.

The System/38 has the capability of attaching up to two system printers. The various models of the 5211 and the 3262 Printers can be attached in any combination, with one exception. System/38 will not support two directly attached A Models.

The printers use a flexible steel print belt that is easily changed by the operator. These inexpensive belts allow the System/38 user to get a range of character sets and print speeds.

Specialized print belts that provide greater speed are available for the 5211 and the 3262 Printers.

4A) 5424 MULTI-FUNCTION CARD UNIT

This optional card device allows users to grow to System/38 from systems equipped with a 96 column card input/output device with a minimum amount of conversion effort.

As an input device, the 5424 can read cards from either of two hoppers and stack cards into any of four stackers. As an output device, the 5424 can punch, print and stack cards in any of four stackers.

Two models of the 5424 are available:

Model A1	Model A2
Reads 250 cards per minute	Reads 500 cards per minute
Punches 60 cards per minute	Punches 120 cards per minute
Prints 60 cards per minute	Prints 120 cards per minute

7D) 3410/3411 MAGNETIC TAPE SUBSYSTEM

The 3410/3411 Magnetic Tape Subsystem offers an optional feature on System/38 consisting of one 3411 Magnetic Tape Unit and Control (both units are in the same frame). One to three additional 3410 Magnetic Tape Units can also be attached.

There are three models of the 3410/3411 subsystem:

- The Model 1 subsystem moves tape at 12.5 inches per second resulting in a maximum data rate of 20,000 bytes per second.

- The Model 2 subsystem moves tape at 25 inches per second resulting in a maximum data rate of 40,000 bytes per second.

- The Model 3 subsystem moves tape at 50 inches per second resulting in a maximum data rate of 80,000 bytes per second.

The 3410/3411 Magnetic Tape Subsystem can operate in two density modes:

- 1,600 bits per inch, phase encoded single density

- 800 bits per inch, NRZI (non-return to zero inter-change on 1's recording), dual density or 1,600 bits per inch, phase encoded, dual density

Both the 3410 and 3411 are table high units with tape reels installed horizontally rather than vertically. A transparent sliding cover permits easy access to the tape reels.

7C1) 3742 Dual Data Station

Used to record keyed data onto a magnetic disk by manual operation of the keyboard. Also, to verify data that has been previously keyed. Disks are read into a 3747 Data Converter to convert captured data onto one-half inch computer tape for subsequent processing. Diskettes can also be read into a System/370 model 115, 125, 135, 145, 155 II, 158, 165 II, 168 via the 3540 Diskette Input/Output Unit. Diskettes can be read directly into System/32 or a System/34, or a System/38.

The 3742 includes two independently operating stations housed in a single physical unit...80 character fixed record size...verify, production statistics, auxiliary duplicating, and six program levels standard per station...120-character CRT display.

The optional 128 Character feature replaces the 80 character record with a record length that is variable from one to 128 characters.

Has buffered storage area into which data is keyed prior to recording on disk, thus allowing for correction of detected errors before record is written. Six program levels are standard. Programs control the automatic functions of skipping, duplicating, field definition, etc. Automatic program selection provided by program chaining enables operator to key up to a 480 character logical record. Modes of operation (ENTER, VERIFY, UPDATE, or SEARCH) are under keyboard control.

CRT Display: Up to 120 characters can be displayed to each operator in three rows of 40 characters per row. The first row displays machine status (mode of operation, column indicator and error codes), rows 2 and 3 display either the data being keyed or the active program, at operator's opinion. Data is displayed progressively as it is keyed to build full records for visual verification.

Magnetic Disk: Diskette 1 storage capacity to record up to 1,898, 80 or 128-character records... Removable and interchangeable among data stations. All data is recorded serially on the disk as standard EBCDIC code.

Standard Features

Verify: Provides the capability to check the accuracy of prerecorded records. If any recorded data is changed as a result of verification, the disk record will be updated upon completion of record...fields that do not require verification may be bypassed by program control.

Production Statistics: Provides machine statistics for use in measurement of workload or production, analysis of errors, and job accounting. Counts automatically under machine control (no programming) data and functional keystrokes, verify correction keystrokes, and records processed. Displaying and/or recording of totals on disk is under operator keyboard control.

Search Address: Provides a means for direct access to a record that the operator specifies by its track and sector address.

Search EOD: Provides a means to directly access the last record of a data set.

8D11-13, 16)
7A)
9E)

SYSTEM/38 I/O STRUCTURE

The I/O structure for IBM System/38 was designed to achieve three major objectives. The first was to develop a channel architecture which allows model implementation tradeoffs, exploits current LSI technology, utilizes the system's virtual addressing capabilities, and allows multiprogramming at the channel program level. The second objective was to decouple the processing unit from the channel by means of a queued asynchronous structure which allows channel program stacking with minimum impact on the processing unit. The third objective was to provide multiple I/O product attachment interfaces for flexibility of added features and to accommodate user migration.

There are two views of input/output apparent to the System/38 user. The first is at the data management level. This level provides device and data independence. Input/output managers (IOM) support that data management level by translating data management I/O requests into channel programs. The second view is at the physical attachment level, that is, the external interface. This physical level provides a number of unique machine (UMI) and multimachine (MMI) interfaces. These two user views of input/output are combined by means of an internal structure. This structure consists of:

- A queued asynchronous system channel boundary.
- A channel processor which executes channel commands or channel programs (multiple commands), allows direct memory access, multiplexed I/O, and supports intelligent I/O adapters (IOA) via a common channel bus.
- Internal IOAs which give to system designers the capability of distributing function from IOM components to an IOA.

IBM System/38 is a virtual storage machine with auxiliary storage consisting of multiple spindles of integrated, nonremovable disk storage. Data, in 512-byte blocks, is paged in and out of main storage across the system I/O channel.

The design approach for attaching the virtual storage subsystem to the I/O channel is described in this article. The high data rate of the disk and the hardware cost of the attachment were the

prime reasons for developing a non-microprocessor design. The disk storage attachment on System/38 handles the function necessary to attach multiple spindles to the system I/O channel.

The attachment incorporates the concept of a shared function controller, whereby each major function has its own separate sequence controller. The term "controller", as used, is defined to mean a sequential state machine designed with logic circuits as opposed to a microprocessor. These controllers are built using high-density programmable logic array (PLA) technology. The ability to subdivide the function into manageable pieces capable of being shared by all attached spindles made this approach feasible.

The attachment controls the disk storage for System/38, performing read, write, and diagnostic functions. The operation for each spindle is specified by an eight-byte command element containing the command, a starting address, and the number of blocks of data to be transferred. Six registers in the virtual address translator (VAT) are allocated for each spindle to point to command and data locations in main storage. The storage management function initializes these registers and issues a channel operation to inform the attachment that the command is ready to be obtained and operated on.

The connection between the attachment and the spindle consists of a bidirectional byte bus used for access and diagnostic sensing operations and a serial data bus. The attachment initiates the access, determines successful completion, and performs the serialization and deserialization functions and the cycle redundancy check (CRC) function.

The attachment is divided into function controllers. The channel controller, access controller, rotational position sensing (RPS) controller, read/write controller, and serializer/deserializer (SERDES) controller each control a portion of the data path and have access to the random access memory (RAM).

All data passes through the RAM, allowing access to the necessary information by each function controller. The RAM is divided into sections which contain command blocks, status blocks, and channel control blocks for each spindle. There is also an 8-byte ID block, an 8-byte header data block, and a 256-byte data block which are shared by all four spindles.

The five function controllers and RAM are shared among the sequence controllers, one for each spindle. These sequence controllers are responsible for determining the proper action of each function controller and locking out each other while using a function controller. The four sequence controllers are located in one PLA. No data flow is associated with these controllers. A set of defined states exists within each sequence controller to represent the allowable command states as defined for the attachment.

8E) Support For Key System Functions

The System/38 machine was designed to support a usage environment characterized by a dynamically changing application load consisting of a wide variety of application types--all utilizing advanced functions such as data base. For example, batch, interactive, and transaction processing, along with program development activities, may all be executing concurrently with dynamically changing workloads and priorities. One of the key requirements for the System/38 instruction interface was to provide efficient support in this type of environment for application requirements such as multiprogramming and data base operations. This centralization of function in the machine simplifies the user programming task and reduced overhead in a dynamic multi-user environment.

Two examples of this system function support will be described here, multiprogramming and data base. Similar high levels of machine capability exist in other major functional areas such as I/O.

System/38 supports multiprogramming through the concept of processes. A "process" is similar to a task in other systems and is the basis for managing work in the machine. The user of the System/38 instruction interface controls the number of processes currently initiated, the priority of each process, and the relationship of one process to another, that is, with respect to processor utilization and storage utilization. The machine then allocates the processor and storage resources based on these parameters as well as on the current status of the process, for example, waiting or dispatchable.

This level of multiprogramming support in the System/38 machine offers advantages like these:

- A single resource management mechanism is applied to processing across all system activities. This reduced overhead and results in better management of resources in a complex and dynamic environment.

- Other efficient resource management mechanisms can be used to take advantage of hardware characteristics without programming dependencies.

Similarly, the System/38 machine provides the basic functional building blocks for a high-function integrated data base. Data base objects include a comprehensive set of functions supporting different access mechanisms, file sharing, record format definition and mapping, efficient record retrieval, update, add, and delete. This allows, for example, a data base file structure to be defined that maps a single physical file into records with multiple formats and content. In addition, a single physical data base file may have multiple indexes (access paths) defined over it, all of which are concurrently updated when the file is changed. Each user of the file may view the data in the form suitable to a particular application.

Overhead Considerations

One of the major problems inherent in the implementation of a high-level instruction interface such as that provided for the System/38 is overhead. In order to reduce the potential overhead, and also to facilitate future extensions, the System/38 instruction interface definition does not require a directly executable implementation of the instruction interface. The instructions and the operand definition dictionary are presented to the instruction interface and are translated into an executable microcode structure called a program object. The internal microcode format is not apparent at the interface.

Having an executable program creation step allows the system to have the advantage of both a high-level instruction interface and reduced overhead at execution time.

In addition, direct support of high-use functions in the System/38 instruction interface, as previously described, is itself an approach toward reducing system overhead. A single implementation of a complex function that can be applied system-wide reduces overhead.

Also, by implementing these functions in the machine, hardware facilities can reduce the overhead that is associated with the higher level implementation typically required in programming.

Summary

The IBM System/38 provides a new type of machine instruction interface that comprises a high level of function together with structures similar to high level language structures and includes computation, addressing, and such traditional programming functions as process (task) management, resource management (storage and processor), data base management, and device handling. This new machine was designed to satisfy major design objectives for the entire system--hardware, microprogramming, and program products. The concept of a high-level machine has been discussed in the literature and has been experimented with in both industrial and research environments; however, System/38 is the first IBM system to bring the advantages of a high-level machine to the business user.

SYSTEM/38 SOFTWARE

SOFTWARE SUMMARY

This section describes major groupings of data processing function provided by System/38.

DATA BASE SUPPORT

8E) 9E) (INTEGRAL PART OF THE CONTROL PROGRAM FACILITY)

A data base is a collection of data that is important to an organization. This collection of data could be kept in any form or on any media.

By this definition, the data stored on any computer system is essentially a data base. However, on previous systems, the data available did not satisfy some of the users requirements. Some data was needed in a different format with different sequences at the same time.

System/38 is designed to solve these problems by making stored data available to anyone who needs it. System/38 Control Program Facility provides the security function so the user can control unauthorized access. The System/38 data base capabilities include:

8F)

- Shared files. Multiple batch and interactive users can simultaneously access, update, add to and delete records in the same file. To provide integrity, an individual record is locked (unavailable to other programs for update) while being updated by one program.

- Externally described data. All data is known to the system. Definition of unique physical files can occur down to a field level. Each field can be completely described through the Data Description Specification with a name, attributes, a text description, an edit code to be used when displaying the field and validity checking specifications that can be applied when the field is entered.

- Multiple access paths. A variety of methods are provided for accessing records in the data base. Programmers need not set up their physical data files in direct, consecutive or indexed organizations. Instead, they first define their data and then must specify one or more access paths to the same data. This method substantially simplifies application design problems. The control program automatically updates all access paths defined for a file.

System/38 data base support starts with the use of the Data Description Specification. The Data Description Specifications enable the user to define data used in the data base and the displays and printers.

These specifications are independent of any specific user program. Therefore, the user can define a field once and it can be referenced by any number of files that are used by programs without subsequent redefinition.

8F2) Data Base Files

There are two types of files that can be defined for the System/38 data base:

1. Physical files. A physical file is a data base file that actually contains data records. The System/38 physical file is similar to disk files on previous systems. All the data records are fixed-length and have the same format. Records exist in a physical file in arrival sequence.

2. Logical files. A logical file is a data base file through which data from one or more physical files can be accessed in a format and organization that is different from the physical representation of the data base.

   Any number of logical files can specify the same physical file as their source of data. Each of these logical files can define a different access path (index) for the data in the physical file, and exclude and reorder the fields defined in the physical file.

   A logical file differs from a physical file in that a logical file has no data within it, but rather a definition of how to retrieve and format fields from one or more physical files. The logical file contains a record format that can:

   - Specify different field attributes than the fields in the physical record (data type, length and decimal position).
   - Exclude fields contained in the physical record.
   - Rename fields contained in the physical record.

- Create new fields by concatenating various fields from the physical record.
- Reorder fields defined in the physical file.
- Select or omit any physical record from access.

This capability, along with the ability to define multiple access paths, is essential in allowing each user in a multiuser environment to view data in terms of his/her own application needs without requiring redundant copies of the same data.

Any changes to a field within a record format require recompiling only those programs using that record format. This is a significant aid in reducing the number of compilations required when data base changes occur.

8F) Device Data Management

Device Data Management consists of two major areas:

1. Display (workstation and system console) Device Data Management which generally uses externally described data (defined by the use of the Data Description Specifications).

2. Non-display Device (diskette, card and printer) Data Management which generally uses program described data. The printer can optionally use externally described data.

8F2) 9) Display Device Data Management. Display devices, such as the workstations and the system console, require the programming of much more function than card readers, printers or a data base file. For example, in addition to the normal information (such as field names, length, data type and decimal positions) for a field on a display device, the programmer must specify:

- Position of the field on the display
- Type of field (input, output or both)
- Additional field attributes (such as highlight, underline, blink, reverse image and protected)
- Output editing (such as decimal point and comma)
- Input validity checking (such as numeric only, mandatory entry, mandatory fill, self-check digit, range check and list of values)

Can this be done on line through screen development

- Command key and other special key specifications
- Display error messages
- Output and input of multiple, repeated lines of data
- Various input/output operations to the device
- Many other functions such as duplicating fields, overlaying information on the display, erasing a portion of the display, positioning the cursor, and keyboard locking and unlocking.

All of these functions are provided by System/38 Display Device Data Management, thereby greatly reducing the amount of coding required to write workstation applications. Most of this function is specified in external Data Description Specifications which are stored in a display-device file.

System/38's device file descriptions let the programmer produce display screen definitions that can be used directly by RPG III programs. Therefore, the programmer defines the display screen once and does not need to redefine that display in the input or output specifications of the application program.

Diskette, card and printer Device Data Management. With these devices, the device file generally describes only the file characteristics. The data fields are described in the user's program. The system allows a considerable amount of flexibility in switching the input or output of a program from one device type to another. For example, output written for the diskette can be printed. All of the special file characteristics can be specified in the device file.

2D) Utilities
9)

Interactive Data Base Utilities (IDU) - 5714-UT1 is a licensed program comprised of:

- Source Entry Utility (SEU) for creating and maintaining source program statements and Data Description Specifications.
- Data File Utility (DFU) for creating and maintaining data files and inquiry into data base files.
- Query for extracting and presenting information from data files.

2D) Source Entry Utility (SEU)

The Source Entry Utility aids the user in entering and maintaining source statements for RPG III, Control Language, Data Description Specifications and utility description specifications.

From a display workstation, the programmer may enter source programs including RPG III or Control Language, and have each statement or command syntax checked as it is entered. For Control Language commands, the programmer may elect to either enter them directly, select from a menu of options or be prompted for each entry.

Note: RPG III compiler is required for RPG III syntax checking

2D) Data File Utility (DFU)

DFU provides these functions:

- Data entry
- Data verify
- Inquiry
- File maintenance

The Data File Utility aids the user in the entry and updating of data file records. Menu and prompt displays are available to guide new workstation users to perform data entry simply and rapidly.

The inquiry capability of DFU supplies a simple, direct method for reviewing specific data.

2D) Query Utility

The Query Utility permits the user to select data from the data base and specify the form in which this data is to be displayed and/or printed. Query differs from inquiry primarily in that inquiry is essentially the selection of a single record, whereas query results in the selection of a set of records and includes the ability to influence the order of the set and generate summary reports.

8C3) Conversion Reformat Utility

The Conversion Reformat Utility operates upon data from a data base file or a device file to perform sort, merge, and copy operations. The conversion Reformat Utility allows the user to:

- Sort a physical file to produce a record address file
- Sort a physical file to produce a physical file
- Sort/merge multiple files to produce a physical file
- Copy data from one or more files to produce a physical file or device file

8D) 4C)

CONTROL PROGRAM FACILITY

(VERSION 1, AVAILABLE AUGUST, 1979)

2D) Spooling

System/38 spooling support is available for input and output devices including workstation printers.

Spooling has the ability to handle output that needs special forms, print belts or alignment aids. Spooling can be used to print many copies, or it can stop and restart printing of an output file.

With spooling, the user can generally decrease the length of time a job runs in the system and increase the number of jobs that can be run in series while optimizing input and output device throughput.

4B) Security
8F2)
1E) System/38 can control access to all data via a user password. Users of individual workstations can also be defined. Security is defined and controlled by the person designated as "security officer," who has full authorization to any object in the system. Security is maintained through user profiles. It may be defined at the object or file level. Security can be specified at the field level by defining a specific or separate logical format.

2D) Copy
8C1)
8C2) This is a function for moving data between combinations of
8E3) devices or data base files. Unlike previous copy utilities, the System/38 Copy function is field-level sensitive. Utility programs can run in a normal job mix along with user programs.

The Copy function gives support to data interchange for the System/38 Diskette Magazine Drive.

2D) System Specialization
9

The unique characteristics of the system (optional devices - workstations, printers, card readers) are defined through normal CL. Further tailoring may be done based upon the use of defaults and specific application requirements. This is used in lieu of a system generation.

2D) Message Facility

The message-handling function of the Control Program Facility (CPF) allows either automatic responses or operator responses

to messages. Operator initiated messages can be sent from any workstation to any other workstation, including the system console.

Pre-defined messages can have two levels of text. The first level is a brief text. The second level is additional text to be displayed, if requested, by the receiver of the message. The system operator and the workstation user can request the second-level message text by pressing the HELP key.

Recovery

Recovery must always be considered in a good system's design, whether the system is batch or a multiprogramming, on-line system. System/38 provides facilities for recovery in both environments. These facilities include:

- General system functions which can be used in recovery design.
- General system functions which can be used in actual recovery.

System/38 functions can assist the user in developing and using recoverable applications. These include:

- Single Level Storage Management - for easier application design and less concern over program size.
- Control Program Facility - handles single-program failures without interrupting the system and will attempt to provide file availability (cleanup) at re-IPL following an abnormal system termination.
- History log - provides the user with information about job start and completion, device status messages and save/restore activities which can be displayed or printed on command.
- Job log - associated with every job. Information is spooled upon completion, and for abnormally terminated jobs, the reason and formatted dump are provided on a spool file at termination.

9E, 2D)

- Data Base Data Management - provides user-specified exclusive use of a file, exclusive update control, shared update and time-out function to control record lockout time for improved user flexibility and control of different processing environments.

System/38 also provides the user with facilities designed specifically for application recovery requirements. These recovery facilities are:

- Save/Restore
- Force

8E3) Save/Restore
9E3)

The Save/Restore function lets the user create a backup copy of his system programs and files. Because of System/38's storage management and the large capacity of auxiliary storage, large amounts of data can be kept on-line.

The Save/Restore function:

1. Saves objects or a group of objects in a library to the diskette magazine facility.

2. Makes space available that was occupied by data after the data was written off-line, but keeps the object's description and the off-line location.

8E3) Force

The System/38 supports a Force function that allows the user to specify the frequency of writing records from an application program to disk storage. This specification determines the maximum number of records for any file in memory at any point.

The Force function is specified by the user for each file when it is created or for a file override. Values from 1 to n can be specified for each file. A Force value of 1 directs the system to force (write to disk) for every added, updated or deleted record. A value of n forces a write to disk for at least every nth record added, updated or deleted.

9E3)

By using a low-cost fixed auxiliary storage, the proposed System/38 will allow all data elements to be accessible at any time. In an on-line data base environment, it is impractical to remove media modules. System/38 provides a copy or save/restore function that will allow for backup and recovery.

9) On-line Programmer Services

With the System/38 Control Program Facility comes an increase in programmer services that permits programmers to do a substantial number of their functions at a workstation. These functions can be performed at a workstation by using direct-command entry, or by following a system-provided menu and prompt display sequence. Either method allows immediate syntax checking.

8E3) Testing
9)
In order to test an application program, System/38 permits the creation of a separate testing environment that can functionally duplicate the situations encountered in the production environment. Also, the test environment can be used for workstation and system operator training.

Using the control program copy function, programmers can take data from production libraries to establish a test library. Through the use of a Control Language command, programmers can test their programs and have the system protect against the changing of any files that have not been declared part of the test library.

By establishing a test environment, programmers can test their programs against live production data while the system's data management function bypasses actual file updates. If their testing procedures require that they update files, programmers may request that copies of production files automatically be placed in their test library.

9) Debugging

The system provides extensive debugging functions that permit the programmer to stop the program at any breakpoint, examine and modify variables and indicators, perform traces, and create formatted dumps. These functions can be performed on any RPG or Control Language program without including specific debug statements in the source code. Most important, all of this debug activity can be performed concurrently with the normal production work of the system.

8F) Integrated Workstation Support
9)
Integrated Workstation Support includes:

- The Workstation Controller (standard for the first 12 workstations).

- The coding of workstations (both display stations and printers) as I/O devices in RPG III.

- The display attributes that can be standardized by the use of the field reference function of the new Data Description Specifications, which include standard names or labels for each field, as well as editing and validity checks.

- The Interactive Data Base Utilities.

  - Source Entry Utility
  - Data File Utility
  - Query

  These were discussed earlier.

8F2)

- The 5250 Information Display System provides two screen sizes, movable keyboards with three different formats, both single and dual station displays, three printer speeds, and local and remote programming transparency.

- The single-terminal programming concept. The programmer develops the application program as a single-terminal program. Additional terminals may be attached to the application. These additional terminals use multiple, logical copies of the application program.

8A1) Control Language

9)

Control Language (CL) replaces and enchances the facilities provided by the System/3 OCL. CL provides the interface to systems function. It is flexible, containing a series of default options. CL is capable of being invoked by the users at a workstation, with a series of prompts and menus provided for assistance. Also, the CL may be used as a source statement in a CL program or in a batch job stream.

9) Prompts

A prompt is a display that requests information. The prompt display consists of a set of fill-in-the-blanks requests that guide the user in entering the parameter values for a command.

Menus

The CPF provides a set of menu displays that help the user request an operation or command. A menu is a list of options from which a selection can be made.

8B) File Reference Facility

Traditionally, library facilities external to data processing systems were employed to maintain current definitions and organizations of data bases within those systems. These facilities all had the inherent problem of synchronizing the external library with the system's current definitions.

System/38 improves the level of control through Control Language commands that permit a programmer to recall information that previously defined a data base file's definition and organization. The File Reference Facility function lets the programmer obtain reports concerning:

- Descriptions of data base and device files
- Overall content and organization of a specified data library

7A) System Architecture

A primary feature of System/38 is the 5381 unit advanced instruction set which embodies many basic supervisory, resource and data base management functions. As an example of the power, the instruction set includes data base operations that retrieve update and logically order data records.

The 5381 has an object-oriented architecture fundamental to its overall design. Objects are structures such as programs, processes, and data base files, which are manipulated at a logical level through the unit's instruction set. The 5381 manages storage on an object basis reducing user dependence on main storage size, physical disk location, and internal implementation.

Access to objects is machine controlled providing a high level of integrity, automatic serialization of concurrent operations on an object, and effective authority enforcement.

Units of work are managed as independent processes (tasks) which share the machine resources (processor, storage, devices). Interprocess communication is accomplished through queues and event signals. Objects can be locked to control and serialize concurrent access to them by several processes.

All objects reside in virtual storage which is managed by the 5381 processing unit. Objects are allocated space on permanent disk storage and are brought into main storage (as needed) where they may be shared by all processes. Although system performance may be affected by main storage size, applications are not limited in the number or size of objects used. This allows additional disk or main storage to be added without a need to restructure applications.

Input/output operations offer improved device independence through the use of the 5381 device support (source/sink) functions which manage the channel, communications, and other asynchronous hardware operations.

The System/38 accomplishes much of its advanced function using main storage resident microcode. The amount of main storage used depends on system size and configuration, and the number of system functions active at any specific time.

This high level of function is standard on all System/38 models.

8E)
9E)

SYSTEM/38 DATA BASE CONCEPTS

The System/38 data base facility differs from existing data base packages in many basic ways. This describes some of the differences by giving an overview of the major functional characteristics. These characteristics include the System/38's design philosophy, its file structure, its sharing capabilities, its expandability, and its data manipulation capabilities.

Design Philosophy

In past systems, full-function data base packages have been applications built on the machine's operating system and its file management component. This has always caused problems with security, integrity, and performance. System/38 is the first computer system to have a full-function data base facility designed as a part of the basic machine. The data base capability is a primary function of the Control Program Facility (CPF) and is comparable to data-base systems previously available only as applications on more expensive machines. All online data in System/38 is stored, manipulated, and accessed through the data base component. The extensive capabilities of the data base facility are designed to be available to the user at whatever level of function and sophistication is needed. The security, integrity, and performance of the System/38 data base facility were enhanced by the consideration of the data base facility during the entire design of the system.

8D15) Files
9E3)

All online data in System/38 is stored as records in data base files. A data base file has three primary attributes: the format of records within that file, the access path for the records within that file, and the members of that file. A record has a fixed format defining each field and its attributes. The access path for a file defines the ordering of records within that file and provides for either random or sequential accessing of those records. The members of a file are different instances of data sharing the same file definition (the same format and access path definition).

The access path defines an ordering of records either by arrival sequence (order of insertion) or by key sequence. The keyed access path provides comprehensive ordering functions. A key exists for every record addressed by the file. A key is made up of fields from the record and system-generated character constants used to achieve hierarchical or duplicate key ordering. Each field can have ordering attributes applied to it: ascending, descending, absolute value, an alternate collating sequence, etc. Additional LIFO, FIFO duplicate key ordering is specifiable for both intra-and inter-file duplicate keys. A selection feature is provided to allow the access path to address a subset of the records within a file based on field values within the record. Keyed access path maintenance represents some additional overhead on data base changes and, therefore, continuous or deferred maintenance options are provided. To ensure that the access path always represents the actual existing data, the keyed access paths are automatically recovered in case of system failure.

The two types of data base files on System/38 are the physical and logical files. Physical files represent the actual stored data. Logical files provide alternate user views of the stored data to support application and data independence and to avoid redundancy of data.

A logical file provides an alternate format and access path for one or more physical files. The logical file format allows a user to see a view of physical record that subsets, reorders, or changes the attributes of the fields in the physical record. The logical file access path allows the user to see a view of one or more physical file members, and subsets, or reorders, the records in those physical members. This provides the user with an alternate ordering of the records for sequential retrieval, or, for random retrieval, a different key than that defined in the physical file.

A different key definition can be specified for each physical file member addressed and any field in the format may be used.

Another facet of this file-based design is security. Security in the data base facility is by file. When a user creates a file, he is the owner of that file and may specify public, private, or normal authorization. "Public" implies all users have all authorizations on the object. "Private" implies only the owner can currently use the object. "Normal" indicates that only the system default authorizations should be made public. The user may subsequently grant any authorities to selected (or all) users and may transfer ownership to another user. To create a logical file, a user may have the correct authority on each of the physical files referenced. To use a logical file, sufficient authorization must be available for both the logical file and its physical files. Field-level security is supported through use of logical files.

8D16) Sharing

Two of the main objectives in the System/38 data base facility design were to significantly increase sharability and decrease data redundancy. The main thrust of the physical/logical file structure is to achieve these ends. System/38 permits sharing on all levels. One of the major new features of System/38 is that file data definitions are specified to the system and shared. Programs may reference by name a central definition for file, format, and field attributes.

This means that the programmer supplies only the format name and then the compiler automatically retrieves from the system the record definition for use in the compiled program. The programmer may choose to ignore the defined format and may redefine field names or even redefine the entire record format, as was the standard procedure in all previous systems. The use of centralized data definition allows improved programmer productivity, fewer errors, improved file maintenance and growth capabilities, and provides the information needed for query, report generation operations, and field level prompting.

The execution-time sharing of System/38 data is very powerful. Different user programs through different logical files see the same data in diverse ways. To achieve this, the System/38 data base facility makes changes visible to all users immediately, prevents users from updating the same record concurrently, and immediately reflects the change in all access paths.

Access paths themselves may also be shared. Another file type called a derived logical file allows different format definitions while sharing the access path with another file.

Expandability

Through use of the physical/logical file structure and the system-defined data capability, System/38 has made the user's data base flexible enough to meet the changing needs of application programs and users. Below are several examples of this.

All file space expansion for additional record needs is handed automatically by the system unless the user specifies constraints.

New physical or logical files can be created at any time for use by new applications. This has no effect on existing physical files, logical files, or application programs.

Modification of the attributes of a data file will not cause the program to need to be recompiled unless the new definition is not consistent with the program's usage. The affected programs are automatically notified at their next usage to indicate their need to be recompiled.

Data Manipulation

The basic data base facility operations are OPEN, GET, PUT, UPDATE, DELETE, RELEASE, and CLOSE, OPEN and CLOSE connect and disconnect the file and the process. GET locates and/or reads a record from a file and has very powerful search operations. For arrival sequence access paths, the search can be sequential or direct. For a keyed access path, the search can be sequential, keyed, or direct.

For sequential searches, the available options are first, last, next, previous, and same. Next and previous imply the record (as defined by the access path) next to the currently addressed record. Same is used when the previous GET was for position only, and now the record is desired.

For direct searches, the options are to find the nth record in the arrival sequence or to find the record $\pm$ n records from the current position in the arrival sequence.

For keyed searches, GET can position to the record whose key is before, equal or before, equal, equal or after, or after the position indicated by the key supplied by the user. Or, GET can position to the record whose key is next or previous to the position indicated by a leading portion of the key for the record currently addressed, that is, a generic next or a generic previous.

On every GET operation, the arrival-sequence position number of the physical record addressed is returned so that the user can, later, quickly reposition to any previously addressed record. This is very useful for "navigating" through a data base structure and can be used to jump from the hierarchical structure defined by one logical file to a different hierarchical structure defined by another logical file, that is, parent to child and back to a different parent.

PUT inserts a record into a file. UPDATE modifies the contents of a record. DELETE removes a record from a file. All three operations immediately update all the keyed access paths to reflect the change. RELEASE unlocks a record that was locked by GET for modification or deletion.

Summary

The System/38 data base facility is unique because of its physical/logical file structure. It does not conform to the relational, hierarchical, or network data base model, but combines aspects of each. A physical file is similar to a relational file. The requirement that relationships between records (key values) must be stored values in the record is similar to that in the relational model. A logical file is a hierarchical file. The fact that multiple logical file is a hierarchical file. The fact that multiple logical files can coexist and that the user can jump from one to the other provides function comparable with that provided by network models. In combining these features of the classical models in this unique way, System/38 has provided the user with an elegant, easy-to-understand data structure.

The System/38 data base facility has extensive capabilities and is a basic part of the design of the system. The data base facility was designed to greatly simplify the application programmer's job and to provide savings to the user in bringing new applications online and in maintaining existing ones. System/38 and its data base facility represent a significant step forward in bringing this level of function to the diverse System/38 customer set.

RPG III

8A), 8B), 9), 2D)

RPG III builds on the functions of the widely used RPG II 8B) and includes many new language functions, such as:

- Externally described data that procides ease of coding, consistent naming conventions and improved application growth capability.

- Full procedural file specification that allows the user to process the same file in both a random and sequential manner in the same program. This file specification allows many common applications to be developed with greater ease.

- Program structures that do not use the RPG II cycle but allow read, write, and update operations to the data base, and allow special workstation input and output operations to be specified directly in the calculation logic. This program structure provides a more straightforward approach to interactive applications and an alternative for complex batch programs. Hence, a "Standard Programming" implementation is now feasible with RPG II.

- The capability to allow any separately compiled RPG III program to be called by another program and also call other programs along with the passing of parameters. These programs can be written either in RPG III or Control Language, thereby enabling a programmer to easily construct a complete application consisting of RPG III and Control Language programs. There is no requirement to combine these programs in any manner (such as a link-edit operation) prior to execution.

- New logic control operations, such as various forms of the DO and IF operations.

- Data structures to allow related data to be manipulated either as individual data fields or as a group.

- Object modules are reentrant. The programmer writes the program just as if the program is dealing with one user. The system automatically saves data areas and indicators for each user.

Compile Options

- After your program is written and compiled, RPG can provide you with a source and cross-reference listing of the generated RPG source program. The listing is provided only when specified by the OPTION parameter. The default is a source and cross-reference listing.

  The listing includes information on both program described and externally described files.

- Library facilities include, source programs, object modules, procedures, subroutines and cataloged procedures

The description of the parameters follows. They specify whether the following options are to be written on the compiler listing when the RPG source is compiled.

- NOSOURCE: No source input or diagnostic listings are written.

- NOXREF: No cross-reference listing is written for variable data item references in the source data.

- NOGEN: No executable program is generated after compilation time.

- DUMP: The compiler dumps all data areas.

RPG and CPF provide functions that you can use to test and debug the programs you develop:

CPF	RPG
. Test library	. DEBUG operation code
. Breakpoints	. DUMP operation code
. Traces	

The CPF functions let you test programs while protecting your production files, and let you observe and debug operations as a program executes. No special source code is required to use the CPF functions.

The RPG functions can be used independently of the CPF functions or in combinations with them to:

- Debug a program, or
- Product a formatted dump of indicator settings and the contents of fields, data structures, arrays, and tables

Source code (for example, the DEBUG operation code) is required to use the RPG functions.

9E) Test Library

The basic concept of testing and debugging is that of a testing environment. Programs executing in a normal operating environment can read, update, and write records that are in either test or production libraries. Programs executing in a test environment can read records in either test or production libraries, but the programs can update and write records only in a test library.

On System/38, you can copy production files into the test library or you can create special files for testing in this library. A test copy of a file and its production copy can have the same name if the files are in different libraries. You can use the same file name in the program for either testing or normal processing.

No special statements for testing are contained within the program being tested. The same program being tested can be run normally without modifications. All testing functions are specified within the job that contains the program and not within the program.

Testing functions apply only to the job in which they are specified. A program can be used concurrently in two jobs: one job that is in a test environment and another job that is in a normal processing environment.

Testing functions of CPF let you interact with a program as it executes to observe the operations being done. These functions include using breakpoints and traces.

Breakpoints

A breakpoint is a statement number or a label in your program where you want program execution to stop. If you use a statement number, it can be a sequence number that SEU assigns when SEU is used to enter the RPG source program.

When a breakpoint is about to execute for an interactive job, the system displays the breakpoint at which the program has stopped and, if requested, the values of program variables. After you get this information (in a display) you can enter CPF commands to request other functions (such as displaying or changing a variable, adding a breakpoint, or adding a trace).

Traces

A trace is a record of the statements that were executed and, if requested, the values of variables used in the statements.

A trace differs from a breakpoint in that when a trace ends depends on which statements are traced and how many statements are traced. The system records the traced statements that were executed. You must request a display of the traced information. The display shows the sequence in which the statements were executed and, if requested, the values of variables used in the statements.

You specify what statements (and what variables) the system should trace. Also, you might specify that variables be traced only when their value changes from the previous time a traced statement executed.

You can specify a trace of one statement in a program, a group of statements in a program, or an entire program.

Debug Operation Code

You can code one or more DEBUG operation codes among your RPG calculations to help you debug a program that is not working properly. Whenever the DEBUG operation executes, one or two lines of debugging information is provided: the first line contains a list of all indicators that are set on at the time the DEBUG operation was encountered; the second line is optional and shows the contents of the result field specified for the DEBUG operation.

You can apply the CPF testing and debugging functions to programs that use DEBUG operations; a breakpoint can be on a DEBUG operation, and a DEBUG operation can be traced.

Dump Operation Code

You can code one or more DUMP operation codes among your RPG calculations to provide a formatted dump of field contents, data structures, indicator settings, array contents, and table contents.

Considerations for using the DUMP operations are identical for those previously described for the DEBUG operation.

8A3)

Not available. We can assist in the evaluation and one conversion to the IBM 5110.

IE)
4D) Security

In an interactive system, the implementation of controls that ensure data integrity and security become especially important because the work stations provide many points of direct access to the system outside the physical control of the data processing department. Without these controls, the potential for data being misused or destroyed increases, especially when many work station users are using the system concurrently.

For many data processing installations, maintaining the security and integrity of data processing information is a primary concern. The most important concern is integrity: the protection of programs and data from inadvertent destruction or alteration. Security is the prevention of access to or use of data or programs by unauthorized persons. Directly related to integrity and security is the need for user identification: the ability to recognize a system user so that only the facilities and data he is authorized to use are made available to him.

The security facilities of CPF provide mechanisms for user identification and for authorizing user access to specific objects. These facilities allow the system to be tailored to provide the necessary level of security and integrity. In addition, the user identification supported by these facilities can be used to design an application-oriented interface for work station users. The system can be tailored so that each work station user has access to only the system functions, applications, and data that he needs to perform his work.

Object Authorization

Object authorization is the process of controlling which system users are allowed (authorized) to use an object and how each user can use the object. Two basic concepts are involved in object authorization: object ownership and object authority.

Object Ownership

Whenever a system user creates an object, he becomes the owner of that object. Unless ownership is transferred to a different user, he remains the owner of the object until the object is deleted from the system. The owner has complete control over his object. He can authorize other system users to use the object and he can transfer ownership of the object to some other system user. Only the system's security officer has the same control over an object as the object's owner.

Data Rights Control

Data rights control how the user can use data in the object. For example, the data rights might let a user read and update data in a file. Data rights provide additional control over the use of data entries within objects and are granted in addition to the object rights a user has. Each system user should have access only to the functions and data they need to perform their job.

CONVERSION TO SYSTEM/38

IBM

IIB) Test Equipment

System/38 customers will receive 14 consecutive days of on-site allowance. This allowance applies to all components connected to the System/38, but does not apply to any inter-connected system (processor) or components. A limit of six 525X devices per system, and three systems per establishment will also apply. The on-site allowance may be converted to preinstallation test hours at the rate of an additional 15% of preinstallation test hours for each day of on-site allowance.

The location of available test equipment will be at 525 University Avenue in Palo Alto.

		PreInstallation Test Hours
5381	Model 34X	30
5251		4
3262		1
3411		1

CPU Cycle

11B) System/38 On-Site Allowance

The System/38 On-Site Allowance (OSA) plan provides for on-site testing of customer programs. The OSA, together with the System/38 Preinstallation Test Allowance, is made available to provide customers with assistance in program testing and data conversion directly related to the on-order System/38. Provisions of the System/38 OSA are:

- The OSA applies to all System/38 models and associated components installed with the System/38, with a limit of six (6) 525X devices per system and 3 systems per establishment. The OSA does not apply to any inter-connected system (processor). Components installed after initial installation of the System/38 do not qualify for an OSA.

- The OSA will commence on the day, Monday through Friday, following the installation of the System/38 and will terminate 14 consecutive days from this commencement date or the day at which productive use is accomplished, whichever occurs first.

Why so Rigid

- Each day of OSA may be converted to additional preinstallation test hours. Each converted day equals an additional 15% of System/38 Preinstallation Test Allowance. The Preinstallation Test Allowance for each System/38 is based on the configuration of the system ordered.

- If the System/38 is installed on lease or rental, such charges will commence upon termination of the OSA. The purchase option accrual period also commences on that date.

- If the System/38 is installed under an IBM lease contract period, the expiration date of the lease contract period will be extended by the number of days in the OSA.

- The first anniversary date of the lease contract period will be 12 months from the termination date of the OSA.

- The warranty period for a purchased System/38 will commence on the day, Monday through Friday, following the installation of the System/38 and will be extended by a number of days equal to the duration of the OSA. The purchase is effective on the date specified by the applicable IBM purchase agreement, but purchase payment is deferred until the termination of the OSA.

- Rental, lease, or purchase status of the equipment must be declared and the appropriate agreement signed and received by IBM on or before the date of installation of the machine. Under the IBM Installment Payment Plan, the Application For Credit and associated financial data must be received by IBM not more than 90 nor less than 30 days prior to the scheduled date of shipment.

- Purchase of IBM equipment after the date of installation may be made under the Agreement for Purchase of Installed IBM Machines. Purchases of installed equipment, including those made during the OSA period, may be considered the purchase of used equipment for Investment Tax Credit purposes.

- All discontinuance notice requirements and termination charge provisions of the Agreement for Lease or Rental of IBM Machines apply during the OSA period.

- Except as described above, all the terms and conditions of the applicable IBM Agreements will apply to the System/38 from the date of installation for purchased equipment or the commencement date of the lease or rental contract period.

11D) Conversion Costs

1) Reference Section 11B for preinstallation test time and on-site testing allowance.

2) Reference Section 12 (Technical Consultants) for appropriate Systems Engineering services.

3) Approximate start-up costs for the 3742 Data Entry Station and System/38 will be $1,000 (Diskettes, Magazines, Ribbons, and Paper).

4) Reference 2C (Environment) for Customer Engineer's write-up for required facilities preparation.

5) Reference Section 5D for Systems Engineering services.

6) Not clear as to how our secretarial service would be utilized. Clearer definition required.

7) Not applicable.

15. TRADE-IN OF IBM DATA PROCESSING MACHINES

Policy

Subject to the conditions below, IBM will accept customer-owner IBM data processing machines in trade against orders for purchase/rental of other IBM data processing machines.

Each machine against which the Limit of Credit is established must be on order (not installed) as of the date IBM accepts the Trade-In Agreement, and installed as of the actual trade-in date.

Consistent with the terms of the policy at time of quotation, trade-in allowances will be based on the physical condition of the machine at quotation time.

Upon receipt of the trade-in machines and installation of the on-order machines, the amount will be credited to the customer.

Should the on-order equipment be cancelled after passage of title of the trade-in machines to IBM, the trade-in machines, if not previously disposed of, will be returned to the customer. If the trade-in machines are not available for return to the customer, IBM will offer the customer a choice of equivalent machines, if available, or the application of the credit against subsequent billings. Transportation charges for returned machines or their equivalent are payable by the customer.

The City of Santa Cruz will need to submit a written request stating projected release date of the System/3 presently installed.

11.E Conversion References

With the System/38 announcement in October, 1978, and first shipment taking place in third quarter 1979, similar installations converting from a System/3 are not available. Reference section 11C for conversion aids available in going from a System/3 to a System/38.

CONVERSION PLAN

IBM

EQUIPMENT AND SOFTWARE COSTS

16. Equipment and Software Costs

A. Hardware

1) Individual Components - See Page 4-8 (Equipment List) for number of units, model numbers, lease and purchase prices. Unit descriptions and rate capacity may be referenced under Section 7 (Hardware). IBM does not offer a five-year lease term, nor a seven-year purchase plan.

2) Total Hardware System - See Page 3-14 (Equipment List) for total cost of each configuration. Financing alternatives are as follows:

1) A month-to-month rental plan,

2) A 24 - 36 month lease plan depending on the equipment type. For example, the CPU would be a 36-month lease item while I/(equipment is traditionally for a 24-month duration.

Under either rental or lease plans, the City will accrue between 40 - 55 percent of the equipment monthly payment toward any eventual purchase of that equipment.

3) An IBM financed 60-month full payout plan which, in the case of municipalities, is really five one-year renewal purchase options dependent on fiscal year availability of funds and council approval.

4) An IBM financed purchase plan over a 48-month timeframe with no cancellation options.

5) Total purchase payment on date of installation.

	Rental	Lease	IBM Financed 60-Month Purchase	IBM Financed 48-Month Purchase	Purchase On Installation
Down Payment	0%	0%	10%	25%	100%
Approximate IBM Finance Charge	N/A	N/A	7.75%	14%	N/A
Maintenance	Included	Included	Customer Responsibility	Customer Responsibility	Customer Responsibility
Insurance	Included	Included	Customer Responsibility	Customer Responsibility	Customer Responsibility
Personal Property Taxes	Included	Included	Customer Responsibility	Customer Responsibility	Customer Responsibility
Accruals Earned Toward Purchase	45 - 55%	40 - 55%	N/A	N/A	N/A
Termination Charges	None	Approximately 2 Months Rental	N/A	N/A	N/A
Discontinuance Option	90-Day Notice	90-Day Notice	Fiscal Year End	N/A	N/A
Conversion Capability to Another Plan	Yes	Yes	No	No	No
Price Protection Plan	N/A	Yes	Yes	N/A	N/A

Maintenance costs may be referenced in Section 2B (Equipment List). Additional maintenance coverage is described in Section 5 (Sample Contract).

Shipping costs as follows:

System/38	$ 475
5251 (10)	430
3411	88
3262 (2)	398
5256 (2)	96
	$1,487

Scheduled delivery for the City of Santa Cruz's System/38 is August 15, 1980, with a requested ship date of January, 1980. When is this firm *

Pre and post-installation testing costs are defined in Section 11B.

B. Software

1) Individual Components - See Page 4-8 for description of software component, identification number, and lease price. Reference Section 8 for use of software component. Installation, under Systems Engineer supervision, remains the responsibility of the user. Maintenance and updating costs are provided under the lease price. Reference Section 13 (Education) for appropriate training costs. Contract terms are defined in Section 5 (Sample Contracts) under the Agreement for IBM Licensed Programs.

2) Total Software costs are included on Page 4-8 (Equipment List).

C. Vendor Cost Addendum

Approximate cost to attach existing 5424 to System/38 is as follows:

Feature 6500	$2,670
Field Installation	1,210
	$3,880

Costs for cables and connectors for (10) 5251's and (2) 5256's require specific location in order to determine appropriate cost. Customer Engineering will be available for consultation with regards to most cost effective method of installation and required specifications.

*ESTIMATED COSTS

Phase I	Lease	One-Time Costs	MMMC
System/38 Hardware (CPU, 3411, 5424 Attach, 3262, (4) 5251's)	$4,406		
3742	237		
System/38 Software	500		
5424			$176.00
5486			40.00
5496 (1)			61.50
Keylocks, 5714-6003		$ 830	
Print Belt		170	
5424-6500		2,670	
Field Installation Charge		1,210	
Freight In		934	
3742 Supplies (50 1D Diskettes)		200	
System/38 Supplies (50 2D Diskettes, 5 Magazines, 6 Ribbons, 10 Cases Paper)		750	
	$5,143	$6,764	$277.50

*Cables, connectors, site preparation, education costs, and sales tax are not included.

ESTIMATED COSTS

Phase II Additions	Lease	One-Time Costs	MMMC
System/38 Hardware (Enhanced CPU, 2nd 3262 (2) additional 5251's, (1) 5256	$1,614		
Keylocks, Audible Alarm, Forms Stand		$ 184	
Freight In/Out		1,172	
	$1,614	$1,356	

Phase III Additions			
System/38 Hardware (Enhanced CPU, (4) additional 5251's, (1) 5256)	$ 984		
Keylock, Audible Alarm, Forms Stand		$ 264	
Freight In		220	
	$ 984	$ 484	

Shipping charges for CPU upgrade in Phase III not included.

Are there any?

ASSUMPTIONS OF LEASE/PURCHASE ANALYSIS

1. First year in Analysis - January 1980

2. Number of Years for Analysis - 5

3. Phase I Purchase Price - $179,736

   (Purchase price does not include software, shipping, cable, education, or supplies cost; use, or property tax; insurance rate)

4. 10% down payment

5. 60 month loan term at 7.75% interest

6. Zero salvage value at 5-year end

7. Straight line depreciation

The following has been prepared by IBM for illustration of a method of financial analysis concerning the acquisition of IBM products and/or services. You are advised not to rely upon this analysis, but to make your own analysis including verification within your business of any underlying assumptions, such as resale, depreciation, and financing. Any savings you can achieve can only result from proper installation and use of the data processing equipment and programs. This analysis, including estimated cost savings is provided for information purposes only and IBM makes no guarantees or representations with respect to its applicability to your business or accounting methods.

CITY OF SANTA CRUZ

MONTHLY CASH FLOW PROJECTION

60 MONTHS

Lease Monthly Payments	$4,921
If Purchased:	
Installment Payment (1)	3,496
Maintenance	1,124
Property Tax (2)	0
Insurance (3)	0
TOTAL	$4,620
Monthly Savings	301
Annual Savings	3,610

Assumptions:
(1) 7.75% Interest
(2) 0% on Book Value, Year End 1981
(3) 0% Per Annum

CITY OF SANTA CRUZ

PAYOFF PERIOD--54 MONTHS

Lease Monthly Payments	$4,921
If Purchased:	
Installment Payment (54 Months)	3,815
Maintenance	1,124
Property Tax	0
Insurance	0
TOTAL	$4,939

CITY OF SANTA CRUZ

RISK ANALYSIS

	1980	1981	1982	1983	1984
Down Payment	$ 17,974	$ 0	$ 0	$ 0	$ 0
Cumulative Purchase Payments (1)	41,954	83,907	125,861	167,814	209,768
Other Ownership Expenses	13,488	26,976	40,464	53,952	67,440
Loan Payoff*	143,899	111,979	77,497	40,244	0
TOTAL PURCHASE	$217,314	$240,836	$261,795	$279,984	$295,181
Cumulative Lease Payments	59,052	118,104	177,156	236,208	295,260
Termination Charges	0	0	0	0	0
TOTAL LEASE	$ 59,052	$118,104	$177,156	$236,208	$295,260
Resale Needed to Break Even	158,262	122,732	84,639	43,776	-79
Current Gross Purchase Price	179,736	0	0	0	0
Resale % Needed to Break Even	88	68	47	24	0
Profit (Loss) if Resale % is:					
90%	3,500	39,030	77,125	117,986	161,841
80%	-14,473	21,057	59,150	100,013	143,867
70%	-32,447	3,083	41,176	82,039	125,894
60%	-50,420	-14,891	23,203	64,066	107,920
50%	-68,394	-32,864	5,229	46,092	89,947
40%	-86,368	-50,838	-12,744	28,118	71,973
30%	-104,341	-68,811	-30,718	10,145	53,999
20%	-122,315	-86,785	-48,692	-7,829	36,026
10%	-140,288	-104,759	-66,665	-25,802	-18,052

(1) Beginning 01/01/80

*See Next Page

CITY OF SANTA CRUZ

PAYOFF CALCULATIONS

	1980	1981	1982	1983	1984
Gross Purchase Price	$179,736	$ 0	$ 0	$ 0	$ 0
Less Accruals	0	0	0	0	0
Net Purchase Price	179,736	0	0	0	0
Less Down Payment	17,974	0	0	0	0
Plus Sales Tax	11,683	0	0	0	0
Net Amount Financed	173,445	0	0	0	0
Interest Charges	12,407	10,034	7,471	4,701	1,709
Principal Paid Off	29,546	31,919	34,483	37,252	40,244
Total Loan Costs	41,954	41,954	41,954	41,954	41,954
Payoff Amount (Amount Financed Less Principal Paid Off)	143,899	111,979	77,497	40,244	0

Loan Starting 1/1/80 for 60 Months at 7.75% Interest

INSTALLED REFERENCES

CITY OF SANTA CRUZ FUTURE APPLICATIONS

INSTALLED IBM REFERENCE ACCOUNTS

City of Farmers Branch - Texas

15-D CCF Shop
Library on Line Remote — 9,000 patrons
Police on Line Remote
Department on Line Local
Tax
Utility - inquiry — 9,000 accounts
Billing - inquiry
Maintenance on line remote is being developed
Financial Control — 18,000 accounts
Interface with Criminal Network
J. W. Wade - Financial Director

Pampano Beach - Florida

Ken Burrough - Financial Director. Completely on line, mark sense for meter readings, financial management using FDP on line, on line offense reports, utility billing, gas pump and library circulations are coming on line now.

Willmington, North Carolina

Larry Blanton - Deputy Director Finance, 919-762-8451

2, #3 warrants, auto and bicycle registration, Soundex System for look up. 4, 7-water sewage refuse, 12, 14, 19, 20 batch, 21-5100, 25-5100, 32-will be soon, 37, 38-5100, 39-on line, walkie talkie, 41.

Battle Creek

Sandy Summers - Data Processing Manager, 616-966-3462

2, 6-tape to service bureau for microfilm, 7, 12, 13, 14, 16.

Police are independent.

City income tax system.

They developed and wrote their own financial system for on line.

IBM

Ada County, Idaho

Mike Bohlan - Director of Data Processing, 208-384-8718

Using 15-D

6-clerk auditor recording office-microfiche, 11-tracing and batch generations, 13, 15-working toward a standardized appraisal system, 21, 40, 41-CCP, 42

Billings, Montana - using 15-D

Director of Data Processing - John Gunthur, 406-248-7511

2, 4, 7, 10, 11, 12, TI mini emulates 3270 controller, 13, 19, 26-partial, 33, 37