U.S. Department of Transportation
Federal Railroad Administration

Railroad Classification Yard Technology Manual

Volume II: Yard Computer Systems

Office of Research and Development
Washington, D.C. 20590

FRA/ORD-81/20.II
August 1981
Final Report

Neal P. Savage
Paul L. Tuan
Linda C. Gill
Hazel T. Ellis
Peter J. Wong

Document is available to the U.S. public through the National Technical Information Service, Springfield, Virginia 22161

NOTICE

NOTICE

Technical Report Documentation Page

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
FRA/ORD-81/20.II		PB82 126806

4. Title and Subtitle	5. Report Date
Railroad Classification Yard Technology Manual-- Volume II: Yard Computer Systems	August 1981
	6. Performing Organization Code U 310019
7. Author(s) N. P. Savage, P. L. Tuan, L. C. Gill, H. T. Ellis, P. J. Wong	8. Performing Organization Report No. SRI Project 6364
9. Performing Organization Name and Address SRI International * 333 Ravenswood Avenue Menlo Park, CA 94025	10. Work Unit No. (TRAIS) 11. Contract or Grant No. DOT-TSC-1337
12. Sponsoring Agency Name and Address Department of Transportation Federal Railroad Administration Office of Research and Development 400 7th Street, SW., Washington, D.C. 20590	13. Type of Report and Period Covered Final Report 14. Sponsoring Agency Code

15. Supplementary Notes

*Under contract to: U.S. Department of Transportation
Transportation Systems Center
Kendall Square
Cambridge, MA 02142

16. Abstract

This volume (Volume II) of the Railroad Classification Yard Technology Manual documents the railroad classification yard computer systems methodology. The subjects covered are: functional description of process control and inventory computer systems, development of computer system requirements, economic analysis, and computer systems acquisition, installation, and management.

Volume I concerns the physical design of railroad classification yards. (FRA/ORD-81/20.I)

17. Key Words	18. Distribution Statement
Railroad yard: computers, process, control, inventory systems, design operations, hardware, cost	Available through: National Technical Information Service 5285 Port Royal Road Springfield, VA 22161

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	126	

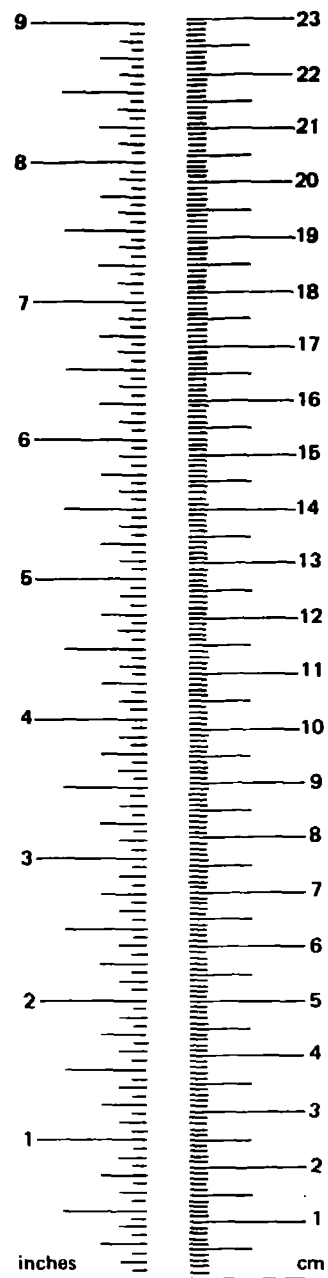
Form DOT F 1700.7 (8-72) Reproduction of completed page authorized

METRIC CONVERSION FACTORS

Approximate Conversions to Metric Measures

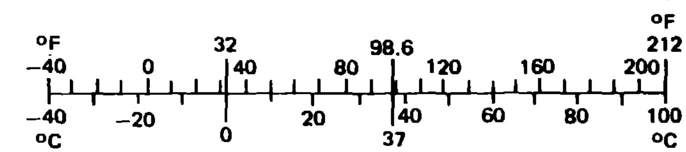
Symbol	When You Know	Multiply by	To Find	Symbol
		LENGTH		
in	inches	*2.5	centimeters	cm
ft	feet	30	centimeters	cm
yd	yards	0.9	meters	m
mi	miles	1.6	kilometers	km
		AREA		
in^2	square inches	6.5	square centimeters	cm^2
ft^2	square feet	0.09	square meters	m^2
yd^2	square yards	0.8	square meters	m^2
mi^2	square miles	2.6	square kilometers	km^2
	acres	0.4	hectares	ha
		MASS (weight)		
oz	ounces	28	grams	g
lb	pounds	0.45	kilograms	kg
	short tons (2000 lb)	0.9	tonnes	t
		VOLUME		
tsp	teaspoons	5	milliliters	ml
Tbsp	tablespoons	15	milliliters	ml
fl oz	fluid ounces	30	milliliters	ml
c	cups	0.24	liters	l
pt	pints	0.47	liters	l
qt	quarts	0.95	liters	l
gal	gallons	3.8	liters	l
ft^3	cubic feet	0.03	cubic meters	m^3
yd^3	cubic yards	0.76	cubic meters	m^3
		TEMPERATURE (exact)		
°F	Fahrenheit temperature	5/9 (after subtracting 32)	Celsius temperature	°C

*1 in. ≈ 2.54 cm (exactly). For other exact conversions and more detail tables see NBS Misc. Publ. 286. Units of Weight and Measures. Price $2.25 SD Catalog No. C13 10 286.



Approximate Conversions from Metric Measures

Symbol	When You Know	Multiply by	To Find	Symbol
		LENGTH		
mm	millimeters	0.04	inches	in
cm	centimeters	0.4	inches	in
m	meters	3.3	feet	ft
m	meters	1.1	yards	yd
km	kilometers	0.6	miles	mi
		AREA		
cm^2	square centimeters	0.16	square inches	in^2
m^2	square meters	1.2	square yards	yd^2
km^2	square kilometers	0.4	square miles	mi^2
ha	hectares (10,000 m^2)	2.5	acres	
		MASS (weight)		
g	grams	0.035	ounces	oz
kg	kilograms	2.2	pounds	lb
t	tonnes (1000 kg)	1.1	short tons	
		VOLUME		
ml	milliliters	0.03	fluid ounces	fl oz
l	liters	2.1	pints	pt
l	liters	1.06	quarts	qt
l	liters	0.26	gallons	gal
m^3	cubic meters	36	cubic feet	ft^3
m^3	cubic meters	1.3	cubic yards	yd^3
		TEMPERATURE (exact)		
°C	Celsius temperature	9/5 (then add 32)	Fahrenheit temperature	°F



PREFACE

This work was performed by members of the Transportation Operations and Information Systems Center of SRI International for the Department of Transportation's Transportation System Center (TSC), Cambridge, Massachusetts. Dr. John Hopkins of the TSC was technical monitor of the project (under Contract DOT-TSC-1337). The effort was sponsored by the Office of Freight and Passenger Systems, Federal Railroad Administration (FRA), as part of a program managed by Mr. William F. Cracker, Jr.

The research was performed under the supervision of Dr. Peter J. Wong, Director, Transportation Operations Research Department. Mr. Neal P. Savage was the project leader. Dr. Paul L. Tuan provided expertise in the areas of process control systems, computer configuration designs, and management structure considerations. Ms. Linda C. Gill and Ms. Hazel Ellis provided technical assistance in the collection, analysis, and synthesis of computer systems literature on process control systems and computer systems acquisition and implementation.

Appreciation is expressed to the following for their cooperation and inputs: Mr. John McGinley (Potomac Yard, RF&P), Messrs. Dean Johnson and John Clendenon (Southern Pacific), Don Hatcher and Jack Lewis (Union Pacific), Phil Seifert (Missouri Pacific), and Larry Trenga and Thomas Levine (Westinghouse Air Brake Company).

CONTENTS

ILLUSTRATIONS

EXHIBITS

Preceding page blank

TABLES

Preceding page blank

1.1 BACKGROUND

The purpose of a railroad classification yard is to sort (switch) cars from inbound trains into classifications (blocks) and to build outbound trains by suitably grouping specific classifications of cars. The classification process entails considerable physical movement of trains, cars, engines, and crews between receiving tracks, classification tracks, and departure tracks. Associated with this physical movement of cars are the processing of paperwork on each car (e.g., waybills), an inventory system to monitor the location of each car, and the control of the routes and speeds of engines and cars.* These management and control functions have evolved from labor-intensive manual systems to highly automated computer-based systems that can greatly increase the throughput and productivity of the yard.

A recent study (Petracek et al., 1976) indicates that over the next 25 years major rehabilitation projects will be carried out at 200 classification yards. At many of these yards, the modernization plan will include the installation of new yard computer systems to replace current manual systems or to upgrade existing computer systems. When both hardware and software costs are considered, the computer systems for a yard represent a major capital expenditure.

In view of their influence on yard throughput and productivity and the capital investments in them, yard computer systems are an important aspect in the design of classification yards.

1.2 PURPOSE OF THIS MANUAL

The operations of a classification yard are complex, involving the operations and coordination of the terminal superintendent, trainmaster, train dispatcher, crew dispatcher, yardmaster, clerks, engine crews, inspection crews, and others. The operations of a particular yard reflect the railroad's management philosophy, local labor agreements, and the historical pattern of performing work at the yard. Thus, computer systems designers often have difficulty in understanding the operational needs in a yard.

Computer systems, too, are complex, and designers often must make sophisticated trade-offs in hardware and software design options. A person not trained in computer technology cannot make informed decisions easily.

*This volume assumes that the reader is familiar with classification yards and their operation. Appendix A is provided for readers who do not have this familiarity.

Consequently, yard computer systems should be designed by a team comprising people experienced in yard operations, who provide user specifications, and computer specialists, who design the computer system that meets these specifications. Because the operations personnel and the computer specialists view the problem from different perspectives and training, however, communication between them can sometimes be difficult. This may result in a yard computer system that does not have the features the user wanted and/or a system that has features the user does not want. Such a system is not easy to use, does not provide the information or performance desired, and can be more expensive than necessary.

These problems can be ameliorated if operations personnel become more familiar with computer systems technology and if computer specialists become more familiar with the operational requirements of classification yards. The purpose of this manual is to provide operations personnel and computer specialists with that familiarity by detailing practical guidelines, procedures, and principles for use in determining the requirements, specifications, and design of yard computer systems.

1.3 FOCUS ON YARD INVENTORY AND PROCESS CONTROL

Included among the many computer systems in a classification yard or terminal are those for:

- Process control
- Yard inventory
- Waybills
- Accounting
- Payroll
- Demurrage
- Crew scheduling.

Treating all these computer systems is beyond the scope of this manual. Rather, attention is restricted to the process control and yard inventory systems and their interface with the systemwide car location and car record system. The process control (PC) computer system controls the speed of cars in a hump yard and routes cars to the proper classification tracks by automatically throwing switches. The yard inventory system monitors the location of cars and trains as cars are moved from track to track, monitors the status and availability of tracks, stores information about each car (the car record) and produces the information and paperwork necessary for carrying out the classification and train-building process. The yard inventory system is often referred to as the yard management information system (MIS).

In this manual, we use the term "yard MIS" specifically for the yard inventory system. This is a restrictive use of the term because it can encompass all the other yard computer systems listed above (except the PC system). Similarly, we use the term "systemwide MIS" specifically to denote the systemwide car location and car record system. Again, this is a restrictive use of the term because such functions as systemwide payroll and accounting could be included in the systemwide MIS.

1.4 CASE STUDY

To assist in the development of the methodology and ensure its practicality, SRI conducted a yard computer system case study for the Potomac Yard of the Richmond, Fredericksburg, and Potomac Railroad (RF&P). The case study effort involved the evaluation of the yard traffic capacity, development of computer systems requirements, analysis of alternative hardware configurations, assessment of benefits from upgrading the computer systems, and recommendations for system implementation and installation. Much of the information and procedures developed in the case study have been incorporated into this manual.

1.5 ORGANIZATION OF THE MANUAL

This volume of the Railroad Classification Yard Technology Manual discusses the use of information and process control computer systems in railroad classification yards. To familiarize railroad personnel with computer systems, Chapter 2, A Computer Primer, describes major aspects of computer hardware and software. No specific railroad applications are discussed.

Chapters 3 and 4 describe the operations and functions of yard inventory systems and process control systems. Each chapter has two major sections, the first describing the operation of the MIS or PC system within the context of normal yard operations and the second describing in greater detail the individual functions of each system. These descriptions cover the input and output of each function as well as the processing completed.

Chapters 5, 6, and 7 discuss design, implementation, and management of classification yard computer systems. Chapter 5, System Design, describes the steps in designing a computer system--survey, analysis, software design, and hardware study. Chapter 6, Computer System Acquisition and Implementation, describes system procurement, software development, and system installation. Chapter 7, Systems Operation and Management, discusses some aspects of ongoing computer systems management.

To present a general discussion of classification yard computer systems, the authors have attempted to select, when possible, representative examples and references from classification yards. Of the many automated classification yards (an installation survey is presented in Appendix B), only a small number have been represented. Yard computer systems vary greatly from yard to yard; therefore, all examples are in some way site specific.

2.1 OVERVIEW

This computer primer is presented for readers who have limited experience with computer systems and for those who wish to review their knowledge of current computer technology and trends.

Unless noted otherwise, the term "computer" refers specifically to stored-program, general-purpose electronic digital computers. A computer system comprises hardware and software: Hardware is the physical equipment, such as mechanical, electronic, and magnetic units; software is the set of computer programs that direct the hardware to complete an application or task.

2.1.1 History

The first generation of computers (1945-1958) began with ENIAC, the first all-electronic calculator. This machine was developed using vacuum tubes for signal amplification and for logic operations. Mercury delay lines and cathode ray tubes (CRTs) were used for memory. ENIAC contained no internally stored programs. UNIVAC I (Universal Automatic Computer) became the first commercially available computer in 1953. This computer contained an internally stored program and used crystal diodes for most logic circuits (Huskey, 1970).

The second generation of computers began in late 1958. The translator replaced the vacuum tube as an amplifier, and magnetic cores replaced mercury delay lines and CRTs as memory. Transistors are advantageous because they are solid-state components that require little power and offer greater reliability than vacuum tubes (Kenney, 1978).

The third generation of computers evolved after 1965 when discrete components were replaced by solid-state circuits. All diode and transistor components were placed on single silicon chips, and the packaging of integrated circuits (ICs) was standardized as the dual in-line package (DIP) (Long, 1977). The IC is sealed in a plastic or ceramic enclosure, and connections are implanted that lead to the enclosure sides.

The single most noticeable advance of the early 1970s was the Intel 4004 microprocessor chip (Long, 1977). Up to that time, each specific application was designed independently, and logic and memory chips and other components were permanently wired together for one application. Instead of fabricating a specific design into the integrated circuit chip, Intel inscribed an entire central processing unit (CPU) on a single silicon chip. When attached to memory chips, the microprocessor (CPU on a chip) became a general-purpose microcomputer, programmable for multiple functions. The microprocessor could be manufactured at a cost of less than $10. When combined with memory chips, a microcomputer could be produced for less than $300. Such a microcomputer is 20 times faster and has 1/30,000 the volume at 1/10,000 the cost of the original ENIAC computer (Noyce, 1977).

Two examples of present chip technology are the chips used in the IBM 4300 series of computers and the Intel iAPX 432 microprocessor. The IBM chips have 132 solder connections, as compared with the normal DIP, which has 16 to 40 connections. These IC chips have 7,000 components with 3-nanosecond transistor-transistor logic (TTL) or 14,000 components with 1.5-nanosecond emitter-coupled logic (ECL). The memory chips in these microprocessors are 18K, 32K, or 64K random-access memory (RAM) with 140/285-, 280/470-, or 440/980-nanosecond access/cycle speeds. This technology allows four 64K RAM chips to be mounted on a 1-inch-square module (Stein, 1979).

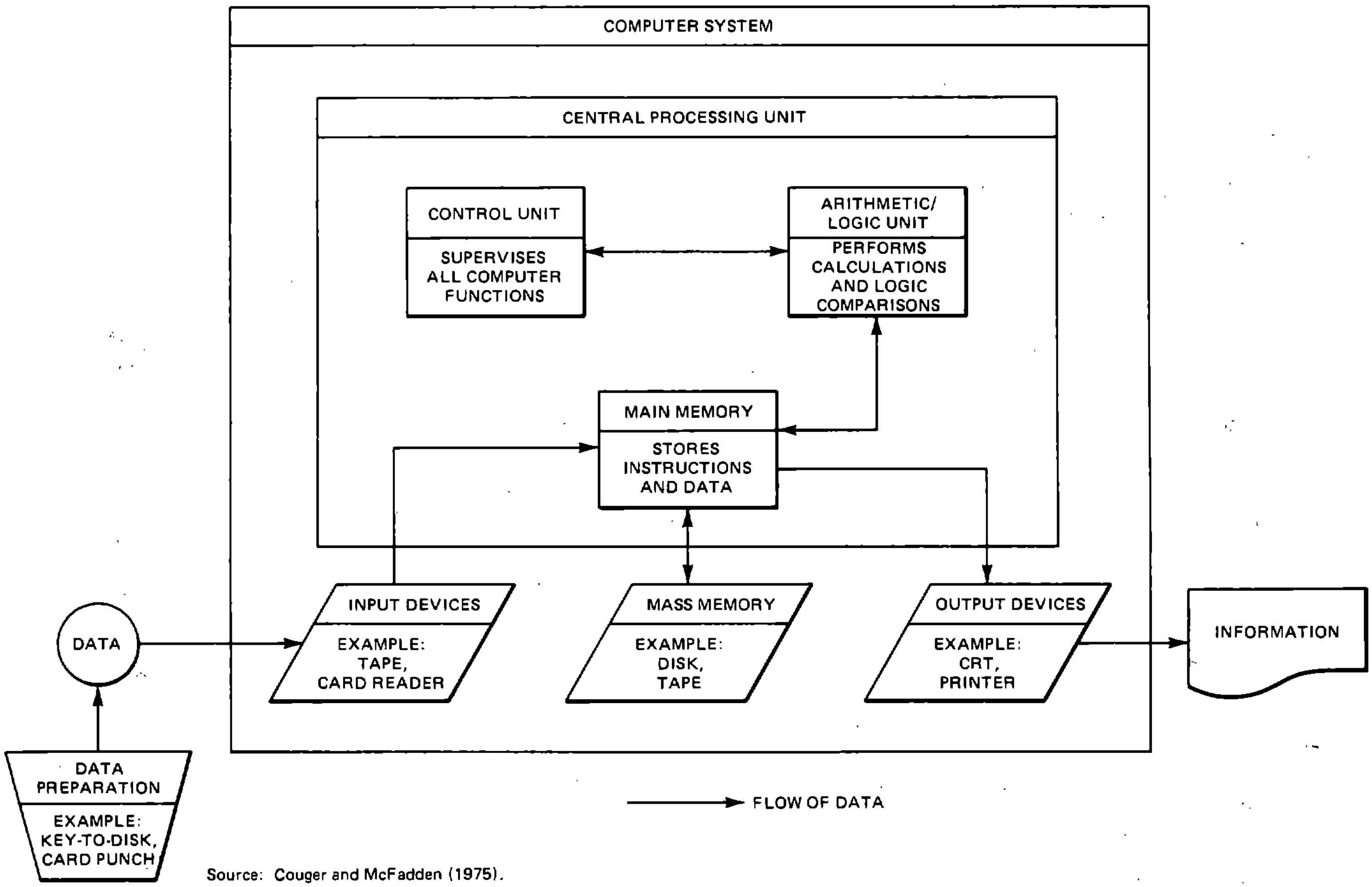
The recently announced iAPX 432 is a 32-bit microprocessor integrating 200,000 transistors. It has a 16 million-byte physical address and has a virtual address space of 1 trillion bytes. The CPU is capable of completing up to 2 million instructions per second, comparable to an IBM 370/158 (IEEE Micro, 1981).

2.1.2 Hardware

The physical equipment that constitutes a computer may be divided into the following functional units (Couger & McFadden, 1975):

- Input devices--Receive data and move it into main memory.
- Storage (or memory) devices--Store data and instructions.
- Control devices--Regulate and control the elements of the computer system.
- Arithmetic/logic (A/L) units--Perform arithmetic operations and logical comparisons.
- Output devices--Transmit or display the results of the computer processing.

These five units are further defined by physical location, as shown in Figure 2-1. The CPU consists of the control unit, A/L unit, and main memory. Mass memory, input devices, and output devices are classified as peripherals. For main memory, magnetic core or MOS (metal-oxide semiconductor) and other technologies are used. Mass memory devices include magnetic disk units, magnetic tapes, magnetic strips, and mass storage



Source: Couger and McFadden (1975).

FIGURE 2-1. FUNCTIONAL HARDWARE UNITS

systems. Input devices include keyboards, punched card and tape readers, and magnetic and optical character readers. Output devices include CRT displays, card and tape punches, and printers. Magnetic tape and disk memories can act as both input and output devices.

2.1.3 Software

The two basic kinds of software are system and application (Huskey, 1970). System software comprises operating system programs that control (operate) the computer, language compilers and assemblers that translate user-generated software into machine code (instructions), and utility routines (system diagnostics, program debuggers, libraries, editing routines, and file management systems). Application or user programs are high-level-language programs that are developed to complete a specific user task. Most users deal primarily with these application programs.

2.2 MAJOR TYPES OF DIGITAL COMPUTERS

Computers are categorized into major types by their generality of application and size.

2.2.1 Generality of Application

The first major feature used to classify computers is generality of application. A computer that is designed or packaged for a limited set of specific uses may be classified as a special-purpose computer. Examples of such computers are PC systems, environmental control systems, electronic games, communications controllers, office systems, and retail sales systems. Computers designed to be applicable for a wide range of uses are classified as general-purpose computers; these computers are often packaged by nature of application. Many manufacturers develop two computers of approximately the same size but for different markets. Scientific machines are designed to process large quantities of numerical data, often in a real-time environment, fed by remote instrumentation. They are programmed in user-oriented languages such as FORTRAN (the most common), APL, and PASCAL.

Commercial data processing systems are designed for business applications. They must be capable of manipulating large data files and of receiving information from many remote data entry terminals. Commercial systems are often used for management information and financial accounting programs and use specialized languages such as COBOL.

2.2.2 Size

Computer size can be measured by price, word size, memory size, processing speed, and functional design. With the dramatic changes in technology and price that occur each year, classification increasingly is based on functional design. [For the last 25 years, cost/performance ratios have improved at an annual compounded rate of 25% (Stein, 1979).] For example, at one time, microcomputers cost between $100 and $1,000 (Kenney, 1978), but with high-volume production the prices have dropped 100-fold. The TMS 1000, the computer industry's most popular 4-bit chip, cost less than $1 in large quantities in 1980. Eight-bit processors such as the Intel 8080 and Zilog Z80 cost less than $2. Newer 16-bit processors such as the Intel 8086 and Zilog Z800, now cost about $200 but the prices are expected to fall quickly.

The minicomputer is a general-purpose computer larger than the microcomputer. It can perform logical, arithmetic, and input/output (I/O) functions under the control of the user. The distinguishing characteristic of a minicomputer as compared with a microcomputer is its larger memory and increased capabilities. A minicomputer can be defined as follows (Kenney, 1978):

- It is a general-purpose computer and not restricted to specific applications.
- It operates through programming.
- It has a versatile I/O capability.
- The central processor costs less than $25,000.

In recent years, a second minicomputer classification was introduced, the "supermini." These 32-bit computers have the capabilities of the mainframe computers of 10 years ago at a fraction of the cost. Representative of these superminis are Digital Equipment's VAX-11/780 and Prime's 750. Superminis currently have a maximum main memory size of 2 million to 8 million bytes, a 600-nanosecond memory cycle, and a cost of approximately $130,000 excluding memory and peripherals (Schultz, 1979). An announced SEL 32/87 32-bit minicomputer is reported to be three times faster than the VAX computer. A 1M-byte memory, a 16K-byte cache configuration costs approximately $235,000 (Coleman, 1981).

The difference between medium and large computer systems is somewhat arbitrary. A medium-size computer can now be considered to be any computer with greater capacity or price than the superminis. Many older medium-size computers may have a greater cost but poorer performance than the newer machines. An upper limit on the medium-size category is the IBM 4341 processor. It is a 32-bit machine with a 2 to 4 MB (mega- or million-byte) memory capacity. The machine cycle time of the IBM 4341 is 150 to 300 nanoseconds and is based on a 64-bit chip. Its purchase price is approximately $250,000 (Blumenthal, 1979). The upper price limit for "medium-size" computers in 1980 was approximately $900,000.

Large computers include the IBM 303X series. The model 3033 has a 4 to 16 MB memory capacity and a 57-nanosecond machine cycle time (Blumenthal, 1979). Recently, larger dual-processor configurations have been announced.

The last classification of computers is a group of special-purpose "super-computers." Pipeline processors use multiple and independent CPUs on a single data stream, and array or parallel processors use multiple CPUs executing the same instructions on multiple data streams. CRAY I and ILLIAC IV are examples of each (Thurber, 1976; Auerbach Publishers, 1981).

2.3 HARDWARE

2.3.1 Central Processing Units

The primary functional unit of every computer is the CPU. The CPU processes data, supervises the various other units of the computer, and contains the main memory. The CPU consists of a control unit, an A/L unit, and a storage unit (main memory). One or more processors act as logic and control units, and an additional processor may provide storage or I/O control.

The control unit interprets and executes the instructions contained in the program in main memory. Data are also stored in the main memory and can be transferred to the A/L unit for processing. The control unit also moves data and programs in and out of main memory.

The A/L unit is used to complete arithmetic operations (addition, subtraction, multiplication, and division) and may make logical decisions for the computer to follow. The time taken to complete A/L unit functions is often used to rate computer productivity or power. Floating-point arithmetic and trigonometric functions are contained either in software programs or, in special cases, in independent microcoded processors. Instructions that are microcoded can be completed much more quickly than the equivalent software execution.

2.3.2 Word Size

Another major characteristic of a computer is its word size. A computer word is the basic internal storage and addressing unit. Word size is expressed as a number of bits. A bit is one switch or memory location and registers either "on" or "off," 1 or 0. Registers usually have a length of one word. Operations in the CPU may use the entire word as one unit or may divide the word into smaller units. The byte (8 bits) is the standard unit used to express one character. Computers with smaller word sizes have less expensive hardware but have limited instruction sets and are also constrained when addressing memory.

The instruction set is the group of instructions to which the computer can respond. One measurement of computer processing efficiency is the "add time," the time needed to complete the add instruction. When comparing the instruction times of computers of differing word sizes, it is important to realize that smaller machines execute smaller, less powerful instruction sets and have limited memory-addressing capabilities.

2.3.3 Main Memory

Data manipulation and logical decision making can only be performed with data and programs resident in the main memory. Main memories require the fast speeds of core and MOS types of memories. The speed of memory is rated in terms of memory cycle time. In recent years, the price of memory has dropped considerably. In 1970, main memory for an IBM 360-65 cost $1.55 per byte. In 1977, memory for an IBM 360-168 cost 27¢ per byte and HP 3000 memory cost 9¢ per byte (Reynolds, 1977). Memory for the IBM 4341, announced in January 1979, costs only 1.5¢ per byte (Auerbach Reporter, 1979), while 64K-chip add-on memory in the HP1000 has dropped to 1¢ per byte (Datamation, 1981).

2.3.4 Mass Memory

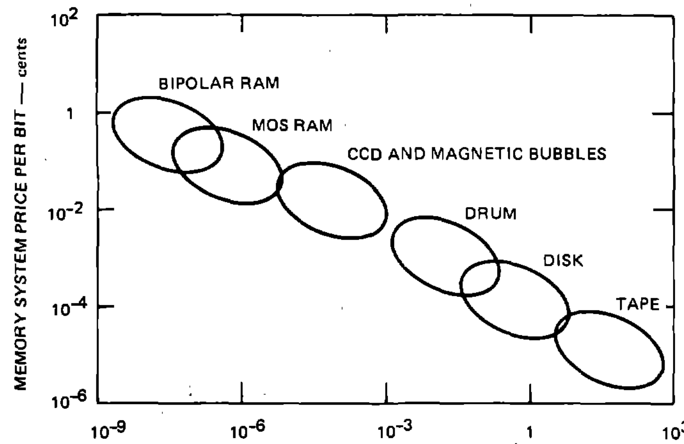
Mass memory allows additional information to be stored and remain accessible to main memory but at a much lower cost than if it were stored in the main memory. Information from these devices is transferred to main memory only when directed and at a slower speed than main memory access.

Secondary or mass memory devices may be categorized as random-access (direct-access) devices or as sequential-access devices. Random-access devices locate information without having to search through the entire file structure. The information is found from an address and therefore does not prescribe a set order to data in the file. Sequential-access devices require a sequential search through the file. This may take minutes as compared with seconds with random-access devices. This relatively slow access time excludes sequential-access devices from most on-line applications.

Random-access devices include fixed and moving-head disks, diskettes (floppy disks), and magnetic drum units. Sequential-access storage devices include tape drives, cassette tape, and cartridge tape. As Figure 2-2 indicates, a direct relationship generally exists between access time (memory cycle time) and cost. Continuing drops in memory prices make cost tradeoffs between semiconductor memory (RAM) and disk memory insignificant in many applications (Business Week, 1981).

2.3.5 Cache Memory

Cache memory is an additional high-speed memory of 1K to 16K bytes (or more) that is used to store the information most likely to be required by main memory. It is used as a buffer to speed the transfer of instructions and data from mass memory to main memory. Memory-accessing routines first search the cache memory before moving to mass memory. If the search is successful (if a "hit" is made), the slower speed accesses to mass memory are avoided. The percentage of hits can be greater than 90% in some applications. For an "average" FORTRAN application using a computer with a memory cycle time of 600 nanoseconds and a cache access time of 80 nanoseconds, the improved memory cycle time is 240 to 400 nanoseconds (40 to 70% hits).



Source: Hodges (1977).

Note: RAM, Random-access memory
CCD, Charge-coupled device

FIGURE 2-2. MEMORY PRICE AND ACCESS TIME COMPARISON (1977)

2.3.6 Input/Output Devices

Numerous devices are available to communicate information to computers and receive it from them. They vary significantly in information transfer speeds, hard copy capabilities, user interfaces, and price. Input devices include punched card readers, magnetic and optical character readers, CRT display/keyboard input devices, key-to-disk systems, paper tape readers, and touch-tone telephones. Output devices include card and paper tape punches; CRT displays; impact, thermal, laser, electrostatic, and ink jet printers; plotters; and microfilm. Magnetic tape and disk units can act both as input devices and as output devices (Couger & McFadden, 1975; McGlynn, 1978).

2.3.7 Interactive Peripherals

Of special interest are I/O devices used for interactive communications. Autotransaction terminals are designed for specific applications, such as verifying credit card transactions or ordering merchandise. They contain a limited number of specified function keys and in some cases a small display. The user cannot change the method or sequence of input (McGlynn, 1978).

A second type of interactive device is the key-to-disk system. The user enters information onto a magnetic disk as queried on a formatted screen produced by the internal processor.

A CRT terminal and keyboard or teletype provide a truly interactive unit, which allows the user and computer to communicate directly. With the addition of memory and a microprocessor, this unit becomes an "intelligent" terminal. The local processor contained in the terminal may provide text editing, input data checking, data packing and formatting, error control, message routing and switching, and in some cases local data processing (McGlynn, 1978). With local memory, a terminal may store 1,000 or 2,000 characters before sending the information to the computer. The computer, therefore, need only communicate with a terminal when it indicates that a message must be sent. This allows the computer to spend less time servicing terminals.

2.3.8 Data Communications Systems

A data communications system is needed whenever data that originate in one location are sent to another location. Terminals, computers, communications equipment and channels, and software form a data communications system (Couger & McFadden, 1975). The five hardware units used in communications systems are terminals, the host computer(s), communications interface devices, communication processors, and transmission carriers (McGlynn, 1978).

2.3.8.1 Communications Interfaces. A communications interface is a hardware component that connects the terminal or computer to the transmission carrier. These devices include modems, multiplexer/concentrators, switching processors, and encode/decoders (McGlynn, 1978).

Modems code and decode digital signals so that voice-frequency lines can carry the information. Modems are classified according to their data-carrying rate, expressed as bits per second (bps) or as baud. Bits per second is the number of information bits transmitted each second. Baud is the number of signal events per second. If one transmission signal contains only one bit, then 1 baud is equivalent to 1 bps. Modems may also be classified as synchronous or asynchronous. Synchronous modems transmit a continuous, uniformly timed stream of data. Asynchronous modems transmit groups of data at random intervals (McGlynn, 1978).

Multiplexers divide the use of communications lines so that a single higher speed line can carry information from multiple lower speed sources. The division of the line may be made using either time division or frequency division methods. Frequency division multiplexing (FDM) divides a channel into a number of discrete, narrow frequency ranges. Each range (subchannel) is used to transmit information simultaneously using the total bandwidth.

Time-division multiplexing (TDM) divides a channel into a number of subchannels using sequential slices of time. Each increment of time is used to transmit part of a different input signal. Each input signal is sampled in sequence until the transmission is completed.

FDM is simpler and lower in cost and is generally used for data rates of less than 1,800 bps. Synchronous time division multiplexers are used extensively for faster bps applications (McGlynn, 1978).

Concentrators are similar to multiplexers, but the use of storage, queuing, and statistical allocation permit use of fewer communication lines for the same volume of input.

Switching processors are used to interface and switch messages between communications devices of various line speeds, codes, and protocols. They may also handle message logging, control, and reporting of terminal and line usage (McGlynn, 1978).

Encode/decode devices encode and decode messages for security and for transmission efficiency.

2.3.8.2 Communication Processors. Communication control may be completed in the main computer or externally. If internal, CPU time and main memory space are needed by the communications control program. A typical communications control program translates code, assembles and disassembles characters, queues input and output messages, interfaces to remote terminals or computers, recognizes and handles various protocols, and detects and corrects errors (McGlynn, 1978).

To free the required CPU and memory, a second programmable computer may be used as a communications or front-end processor. A communications processor replaces the functions of an internal communications program and can also function as a multiplexer, concentrator, or switching processor.

2.3.8.3 Data Transmission Carriers. A transmission carrier is needed to communicate between remote hardware locations. Common carriers, offering the public hire of leased lines, are often used. Services offered are classified by bandwidth, which is the rate at which data may be transferred. Common bandwidth ranges are (Couger & McFadden, 1975):

- Narrow band--45 to 300 bps.
- Voice band--Voice-grade telephone lines to 9,600 bps.
- Broad band--Coaxial cable or microwave to millions of bits per second.

A communications line may be a direct leased line or an ordinary telephone line. Leased lines are purchased at a fixed monthly rate, independent of usage, and offer faster response times, a better quality transmission, greater privacy, and increased flexibility. Other broad-band transmission technologies include satellite transmission and optical fiber transmission (Couger & McFadden, 1975; McGlynn, 1978).

Different protocols (transmission coding) are used to send digital data through transmission lines. They may be classified into two groups, asynchronous and synchronous. Synchronous protocols may be further divided into character-, bit-, and byte-oriented protocols. An asynchronous protocol transmits data at a nonuniform rate, requiring a start and stop bit for each character sent. Because each character is sent instantaneously, no buffer is required at the terminal or other sending device. A synchronous protocol requires that timing signals be sent between devices but provides a more efficient transmission for higher speed devices because no added bits are needed for each character (McGlynn, 1978).

2.3.9 Reliability

System reliability is commonly expressed as mean time between failures (MTBF). For many users, a measure of availability is also important. An estimate of availability (A) for each component can be expressed using the MTBF and mean time to repair (MTTR) (Champine, 1978), as follows:

$$A = MTBF/(MTBF+MTTR)$$

Table 2-1 presents the MTBF and MTTR estimates for various system components. The availability of a system is the product of the availabilities of the individual components on which the system relies. Therefore, as the number of critical components increases, the total system availability decreases. If a system has 20 independent components, each with a 99% availability, the compound system availability would be 0.99^{20} or 82% (Champine, 1978).

TABLE 2-1. - ESTIMATES OF MEAN TIME BETWEEN FAILURES (MTBF) AND MEAN TIME TO REPAIR (MTTR) FOR COMPUTER COMPONENTS (hours)

Unit	MTBF	MTTR
Moving-head disk drive	1,000	1
Fixed-head disk drive	1,500	1
Disk controller	14,000	2
Tape drive	1,000	2
Tape controller	20,000	3
Processor and console	7,700	1.5
4K core memory	34,700	1.5
8K core memory	25,000	1.5
Real-time clock	200,000	1
Multiplexer	40,000	1
Bus switch	20,000	1
Arithmetic unit	200,000	1
CRT terminals	2,000	2
Computer with 28K memory	3,500	2

Source: Kenney (1978).

System failures can be attributed to hardware failure, software failure, operator error, and environmental errors. Software failures are discussed in Section 2.4.6. Hardware failures occur at the component level. To get beyond the high failure rate of early component life ("infant mortality"), most components are run for this period in the factory. Environmentally produced failures occur less often when the electrical and thermal environment are maintained within the optimal operating range with little variance (Champine, 1978).

The method used to provide acceptable reliability depends on the type of reliability that may be attained. Three types are discussed here: The prevention of downtime of any kind, the prevention of prolonged periods of downtime, and the protection of the data base (Yourdon, 1972).

2.3.9.1 Continuous Performance Reliability. In a real-time system requiring continuous performance, no failure of any duration can occur. Systems are therefore judged on their MTBF. Redundancy and multiprocessing are the methods used to provide this type of reliability.

For complete redundancy, the system must have two or more of every critical component. When any critical component fails, the other components continue to operate (Champine, 1978). For fail-safe redundancy, the failure of any one component does not degrade the user response or system capability. Multiple CPUs may be required to operate concurrently and poll each other to detect processor errors. If three or more processors are running concurrently, they may vote to determine the defective component. This method provides the best method to assure data integrity, as affected by processor error. If the backup processor is not used for concurrent processing, it may be used for secondary tasks such as batch processing. When the primary processor fails, the secondary processor is switched to the primary task. A "fail-soft" system provides continuous processing through incomplete redundancy. A fail-soft system guards against complete system failure, but the level of service is generally degraded when a component fails. When a failure occurs, some functions such as batch processing must be eliminated and a delay in response time may be experienced.

A second method to improve reliability is multiprocessing, whereby no backup system is used. A single system is configured with multiple processors, main memory, I/O devices, and peripherals. When a component fails, the system is automatically reconfigured to continue in a degraded mode. If an insufficient processing capacity remains, the total system cannot operate at full functional capacity. In this case, the system sheds noncritical functions until a manageable processing load is reached. This method requires a thorough design of system architecture to ensure that sufficient capacity remains for all critical functions. This is a fail-soft hardware configuration because it provides backup processing but does not provide complete redundancy. It is generally less expensive than full computer backup because smaller computers can be used.

As the number of redundant components increases, the degree of degradation for each single failure decreases (Scherr, 1978). Multiprocessing systems run optimally with multitask processing loads because single failures then only affect independently run jobs on the system. On the other hand, if the system involves a single task, the failure of a single component may prevent completion of the task. Efficient programming is increasingly difficult as the number of multiple processors increases. For a given application and total system size, as the number of processors increases, the size needed for each decreases. As the largest unit of processing power and memory decreases, the effort to modularize the application to that size and to control intermodule communications increases greatly. In addition, as the number of processors increases, the overhead to run them also increases.

At failure, reconfiguration may be by an automatic or semiautomatic switch. The fastest method is to have a second processor connected to the same memory and communications lines as the first so as to perform the same functions in parallel. This configuration is called automatic switchover or hot standby. The primary processor is used as a stand-alone computer to perform all required functions. The secondary processor is always on hot standby, ready to cut over at any time. Inputs are received by both processors simultaneously, and calculations are made in a redundant manner. However, only the output from the primary processor is usually allowed to exit; the output from the secondary processor is compared with that of the primary processor. If a difference is detected that is greater than predetermined tolerance limits, a determination is made about which computer is at fault. The failed computer is shut down, and the other uses the register contents and program pointers of the failed processor to immediately continue processing. Figure 2-3 depicts a typical hot-standby dual-processor system.

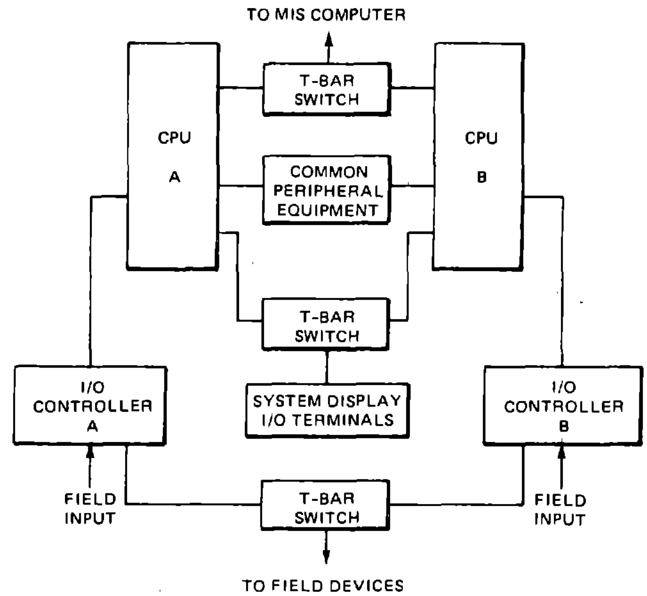


FIGURE 2-3. CONFIGURATION OF HOT-STANDBY DUAL-PROCESSOR SYSTEM

The semiautomatic switchover configuration also consists of two identical computers. The difference is that the change from one to the other is not automatic, and therefore the secondary computer is not required to operate in parallel at all times. When the primary computer fails, the secondary computer must be manually switched into operation. If the secondary computer has access to the files of the primary computer, it can quickly begin operation at the same place. This is usually accomplished by using a shared or switchable disk or by recording data on a transportable medium. One method is to have this disk contain all required files for normal operation. A second method is to periodically update the files of the secondary system with the status of the primary system. Between updates, critical events are stored (checkpointed) to the switchable or transportable medium. Upon failure of the primary computer, the secondary computer first updates its status from the checkpoint file. This method reduces the size of the shared files. The frequency of checkpointing depends on the amount of lost data that is acceptable, the number of disruptions to the system that are acceptable, and whether the cost can be justified. A third restart method is to generate an audit trail. An audit trail records all input messages, which allows the program to rerun from the time of the last valid set of data (Champine, 1978).

If the files of the primary processor are not available, the required software must be available from another source. The secondary processor cannot begin operation from the exact point of failure, and the starting point must be manually entered.

In a cold standby system, the secondary computer may be engaged in a different application while the primary computer is in operation. If the primary computer fails, this secondary processing must stop and its functions must be transferred to a manual backup or remain uncompleted.

2.3.9.2 Prolonged Downtime Reliability. The second type of reliability is the prevention of prolonged periods of downtime. Here, both the MTBF and MTTR are of importance. The MTTR may be decreased by improving preventive maintenance, buying increased or guaranteed maintenance service, programming automatic restarts, or providing backup components. If the frequency of failure increases, the length of acceptable repair time most likely will be shortened because of decreased user tolerance (Yourdon, 1972).

2.3.9.3 Protection of Data. The third type of reliability, protection of the data base, is becoming increasingly important as more users have access to common data bases. Three methods can be used to increase the reliability of the data base (Yourdon, 1972). At fixed intervals, periodic copies of files may be made on a second, more reliable medium such as magnetic tape. In case of a failure, files can be restarted at the point of the last "dump" (copy to tape).

An audit trail may be kept, recording all transactions and retaining a copy of the affected data base records. In case of failure, any uncompleted transactions are reprocessed using the original data record (Yourdon, 1972). A combination of periodic dumps and audit trails is called an incremental dump. In addition to periodic dumps, copies are kept of all file changes since the last dump.

The third method is to record data on two disks. This provides a check of data integrity and a backup in case of failure. This method does not prevent both data bases from being destroyed by application program errors, however.

System integrity is of equal importance to hardware reliability. Errors that generate questionable data must also be identified and corrected. To ensure integrity, error detection and correction or retry must be resident in main memory, mass memory, and I/O and data path areas.

2.3.10 Maintenance

Reliability and availability requirements increase with time because of users' demands for better service. At the same time, the cost of maintenance to provide this level of service also increases. Currently, the charge for a typical maintenance agreement is 5 to 10% of the original hardware cost per year (Gabet, 1979).

Manufacturers are using several methods to provide increased reliability while keeping maintenance costs low. Repairs that can be completed on site save technician time and travel. Redundant hardware allows the system to continue operations until the part can be replaced. Main memories can be installed with an error-correcting code that corrects all single-bit failures and identifies all double-bit errors. This helps the system to continue operations until a technician arrives.

Built-in maintenance processors can automatically identify and log many errors. Often this allows operations personnel to replace the failed component. For more difficult problems requiring maintenance personnel, the maintenance processor can provide quick, accurate diagnostics. Taken one step further, the diagnostic processor can be linked via telephone lines to a remote maintenance computer. This link provides access to special test routines, special dump analyses, a data base of system problems and solutions, and interactive diagnostics by factory experts (Champine, 1978).

2.4 SOFTWARE

Software is the collection of instructions given to a computer to direct it to produce desired results. Covered in this discussion of software are operating systems, language processors, libraries and utilities, data management, application software, reliability and maintenance, and documentation.

Whereas hardware costs have been declining each year, the reverse has been true for software costs. Software is produced by skilled individuals whose salaries have steadily increased. In a new computer system, application software may be 80 to 200% of the hardware and system software cost (Kenney, 1978). As costs increase, manufacturer-provided software becomes a greater factor in total system price.

2.4.1 Operating Systems

Operating systems software controls the internal operation of the computer hardware and software systems. Operating systems have been developed to manage the increasingly complex assemblage of tasks that computers perform. They are designed to allow easy user access to resources and to allocate those resources productively.

First-generation computers allowed execution of one program at a time, and control of the computer was given to an operator or user. As hardware capabilities increased, the percentage of time spent executing (processing) user jobs decreased. System resources were no longer used efficiently because with one user, a large percentage of time was spent in queuing, setting, and controlling jobs. In addition, resources were not used efficiently because with I/O-bound applications, the CPU remained idle during I/O operations.

To alleviate this misuse of CPU time, newer operating systems allow the loading of multiple programs into the system. Multiprogramming permits a second program to use the CPU while the primary program waits for its I/O return. Currently, many processors operate at 1 million to 10 million operations per second but must access disk memories at 50 milliseconds. Consequently, these processors have the power to perform 100,000 to 500,000 operations during the time of a file access (Flynn, 1979).

Operating systems are designed to use the computer itself to relieve the operator of control functions and to provide multiprogramming. They provide a standardized structure with which users can obtain access to the computer. Although standardization imposes some restrictions, it allows much greater efficiency in the execution of processing jobs and frees the programmer from any computer operations responsibility (Sayers, 1971).

2.4.1.1 Features of Operating Systems. In addition to providing for better use of system resources, operating systems integrate features appearing in second-generation computers (Sayers, 1971). Relocatability allows storage of programs in a location other than the beginning of main memory. Thus, multiple programs can be loaded in the computer at one time. A universally used memory control program must reside in memory along with application programs.

Reenterability allows a program to be executed a second time before it is completed the first time. This becomes important for multiprogramming because multiple programs may have to execute the same part of the operating system concurrently.

Interrupts are messages sent between programs within a computer. They alert the control program that external servicing for a program (such as receiving input) has been completed and that the program is again waiting for CPU time.

Channels are small, independent processors that handle only I/O operations, such as data transfer and scheduling and queuing I/O requests. They allow the CPU to remain free for data processing. Interrupts from the channel signal completion of a particular job (Lister, 1975).

A linkage editor/loader formats a program for execution and assigns addresses to the code in reference to the program beginning. When a program is loaded into main internal memory, the linkage editor stores the location of the beginning of the program. Any references to the program are then made through the linkage editor, which can compute real locations by adding the program's beginning location to the relative address in the program.

2.4.1.2 Subprograms. An operating system can be divided into the following three main subprograms (Sayers, 1971):

- Executive subprogram--The highest level of control, the executive subprogram controls the other functions of the operating system. A program or operator communicates to the operating system through the executive subprogram.

- Task/resource management subprogram--The task/resource management subprogram provides for the control of the use of system resources. System resources include main memory, CPU time, I/O operations, and the system clock. These resources must be allocated among the currently waiting programs and the internal operating system tasks. Task management includes execution timing and control, job queuing and scheduling, priority and interrupt recognition, and system error recovery. This subprogram also arranges programs in main memory, brings programs from mass memory, and feeds programs into the CPU for execution.

- I/O management subprogram--The I/O manager schedules and executes I/O tasks. This subprogram must control the space on disks used for direct access, allocate the use of tape drives and printers, provide a directory to external memory, close and open files, and provide I/O error recovery.

2.4.1.3 Types of Operating Systems. Five general types of operating systems exist. A serial batch operating system schedules jobs sequentially, processing only one at a time. Multiprogramming operating systems allow many jobs to run simultaneously. The CPU does not execute each job simultaneously but divides CPU and I/O operations, allowing one job to be executed in the CPU while others wait for I/O returns. To the user, multiprogramming appears to be the same as serial batch processing because he or she may have to wait for entry into the system and once the job is completed output is returned all at one time (Sayers, 1971). Multiprogramming jobs may be submitted indirectly (off-line) using a card or tape reader or directly (on-line) using a CRT display or teletype.

Time-sharing systems attempt to give the user the impression of exclusive use of a computer. Time-sharing users enter and receive data and instructions on-line using a CRT display/keyboard or teletype connected to the computer. Because the speed of human typing is slow compared with normal I/O or CPU processing, the operating system can provide a response that the user perceives as immediate. This fast response time allows interactive user-to-computer communications. Time-sharing operating systems assign each program a percentage of the CPU time and rotate programs in and out according to this schedule.

Real-time operating systems usually run only one program. The program is usually input from one or more locations on a time-critical basis (Lister, 1975). For real-time applications, even the delayed-time operation of a time-sharing system is too slow. A typical real-time application is classification yard control, which requires instantaneous entering and processing of car movement readings to determine car retardation and/or switch positioning.

Virtual memory operating systems are the fifth type of operating system. With multiprogramming and time-sharing, the CPU is used more efficiently but great amounts of memory are required. To extend the relatively expensive main memory, virtual memory systems treat part of disk storage as main memory. The main memory is divided into "pages," usually 500 to 1,000 words long, that are brought (swapped) into main memory when needed.

2.4.2 Programming-Language Processors

Programming languages were developed to provide a method for directing the computer to perform tasks for the user. The most fundamental way is to make a list of the machine instructions that must be performed to accomplish the required task. Because no translation is needed, the CPU may execute the "code" directly.

The early users of computers found machine languages difficult to learn, tedious to use, difficult to change, and error prone (Couger & McFadden, 1975). To facilitate generation of computer instructions, language processors were developed. These programs use the computer to translate a user-oriented source language program into machine-readable binary instructions.

2.4.2.1 Low-Order Languages. Low-order or assembly languages are written to closely resemble machine instructions. Binary instructions are represented with mnemonic codes, and binary addresses are translated to simpler relative addresses and represented by alphanumeric codes.

During the assembly process, each source statement--normally comprising a label, a mnemonic operation code, and an operand (or a variable)--is translated into its machine language counterpart, typically in octal-coded binary form. Because of the one-to-one equivalence between the assembly language and the machine language, the programmer is offered the flexibility of taking advantage of machine hardware features--for example, bit-position manipulation, special logic instructions, and I/O and interrupt devices directly accessed by assembly programming. The assembly language also enables the programmer to economize in the use of execution and memory cycle time and to achieve a better control of memory space use. Compared with machine languages, low-order languages offer easier programming, efficient instruction use, and easier error correction (debugging), and they do not require that the programmer keep track of absolute storage locations (Couger & McFadden, 1975).

Because they offer the user the ability to take advantage of hardware characteristics, assembly languages are suitable for classification yard sensor data acquisition programs in which interface with communication I/O channels is the rule rather than the exception. Assembly languages are applicable to detector scan programs, I/O interrupt routines, and switch and retarder control programs.

Assembly languages are still far removed from normal human logic and thought. Problems to be solved must be broken down into small, detailed increments. Coding and debugging are slow and difficult; and because instructions are machine dependent, programs in assembly languages are not readily transferable to a different manufacturer's computer (Couger & McFadden, 1975).

2.4.2.2 High-Order Languages. High-order languages (HOL) are oriented toward users' problems rather than toward machine hardware characteristics. One statement in an HOL may be equivalent to 10 to 20 assembly or machine language instructions (Couger & McFadden, 1975). HOLs are English- or mathematics-like so as to be easily learned, easily programmed, easily changed, and hardware independent. Because learning and writing HOL programs is easier, the lead time is much shorter than for a comparable program written in assembly language. Standards for HOL compilers ensure that applications programs can be transferred to different hardware and ensure that programmers can work on many different computers without retraining. The relative ease of debugging, modification, and documentation makes HOLs useful for programs that are subject to frequent changes. Therefore, using a language such as FORTRAN or COBOL is usually desirable for complex control algorithms, inventory programs, and other non-I/O-dependent tasks.

HOLs are usually not used for PC applications unless they are extended to include real-time control process instructions that enable the program to interrogate external interrupts and to interface directly with I/O devices and computer timers.

The complexity of HOL compilation (translation) requires additional machine time and memory space and, because of the flexibility of the HOL instructions, compilers may generate some inefficient codes. Despite these disadvantages, lowered programming, processing, and memory costs make HOLs increasingly attractive to most users.

Among the HOLs in general use today are (Couger & McFadden, 1975):

- Scientific use--FORTRAN, ALGOL, PL/1
- Business--COBOL, RPG
- Time-sharing--BASIC, FORTRAN, PASCAL
- List processing--LISP, SNOBOL
- Simulation--SIMSCRIPT, GPSS.

2.4.2.3 MIS Software Language. The software language chosen for use in a classification yard MIS computer system should be well suited to file accessing, file manipulation, and report and display generation. A large amount of scientific processing or mathematical computation is not necessary in management applications.

In general, MIS applications on a large computer fit well with high-level business- and file-oriented languages, such as COBOL. Programs written in an HOL are preferred because they are easier to develop and modify. Programs written in a standard machine-independent language can be transferred to other machines and can take advantage of software developed on other machines. If a microprocessor is used for an application or as a communication processor, a lower level language may be better. Because of the smaller memory and processing power, microprocessors benefit from the efficiency of assembly languages.

2.4.3 Libraries and Utilities

Many computers and operating systems are delivered with a set of independent programs and routines designed to be of general use to most users. Data-handling utilities provide user-callable control of printers, CRT terminals, tape drives, and other peripherals. These routines allow formatting of output, device-to-device data conversion and transfer, and listing of directories, tables, memory addresses, and the like.

Sort/merge utilities are packaged routines that provide for the efficient sorting and merging of files. Sorting routines take strings of unordered records and sequence them according to user-specific keys such as control fields; customer name, part number, and record number; ascending or descending order; and record size. Merge routines take files and combine them in an ordered manner.

Library routines provide an ordering method and locations for storing user-callable routines. Common programs found in libraries include: mathematical functions, trigonometric functions, data reduction routines, and regression analyses. The library maintenance program allows users to add their own routines and programs to the library.

2.4.4 Data Management

A method of organizing, accessing, and updating long-term on-line storage of data files is required for most computer systems. Disk files are needed that allow easy access by all users and that eliminate the need to reenter often-used data and programs via slower speed tape drives, card readers, and CRT terminals (Lister, 1975).

2.4.4.1 File Management Systems. File management systems are designed to create and delete files, locate and access on- and off-line files, allow file references by address-independent symbolic names, permit and restrict the sharing of files, and protect and/or restore files in case of computer failure (Sayers, 1971). Files may be accessed by sequentially searching each file for a label comparison, but most file management systems use a mapping technique. Directories contain a list of symbolic file names and their physical addresses in mass memory.

In many file management systems, a multiple level or hierarchy of directories is used (Lister, 1975). In a three-level hierarchy, a master file directory contains a pointer to each user file directory. Each user directory contains a list of user internal directories that in turn contain a list of file names and their physical addresses. A complete file name contains a part symbolic of its location in each directory level. For example, the file name CHAP.2 in directory BOOK of user SMYTH could be referred to as SMYTH BOOK CHAP.2. In addition, a file name may also contain a reference to its edition, for example, CHAP.2 and CHAP.2 OLD.

A user can add, write, change, and delete files within his or her own directory. When files external to the user are created, they are controlled by the file creator. In most file management systems, the control of file reading, writing, and deletion can be restricted to the user, specified others, a group, or any user, as determined by the file creator. A file management system will usually allow multiple users to read a file simultaneously but will allow only one user at a time to write to a file (Sayers, 1971).

Backup files are often automatically created to allow recovery if a system failure occurs when a file is open. Backup files may be checkpoint files, transaction data files, or previous versions or editions (Sayers, 1971).

2.4.4.2 Data Base Management Systems. Data base management systems (DBMSs) allow the creation of a more complex data structure and file organization. A data base has been defined as:

> ...a collection of interrelated data stored together with controlled redundancy to serve one or more applications in an optimal fashion; the data are stored so that they are independent of programs which use the data; and a common and controlled approach is used in adding new data and modifying and retrieving existing data within the data base. (Martin, 1976).

A DBMS is a set of programs and user-available routines or commands that create and maintain a data base, retrieve data, and produce output data. Data base creation and maintenance entail file structuring, input transaction processing, logical and interactive maintenance, and file reorganization (Sayers, 1971). The simplest file structure is a sequential one. Records in the file are not stored in any order, and the user searches for a particular record by sequentially passing through the file.

In a hierarchical file, the file structure resembles a family tree, with each record (parent) pointing to a number of lower level records (children). Each child in return may act as a parent to lower level records. If a structure is defined such that a child may have more than one parent, the structure is called a network or plex structure.

An indexed structure reserves a portion of the file for storing keys that index information in the main body of the file. In a list structure, each record contains a pointer to the address of the next successive record. If the last record of the list points to the top of the list, this is a cycle or ring structure (Sayers, 1971).

A relational data base uses a simpler method of logically representing hierarchical data structures. Pointers to multiple records are eliminated. To link children to parents, the identifying field of the record is a combination of the identifying fields of the parent and child (Martin, 1976).

These file structures refer to the file organization of the data base, not to the structure of user-accessible files of the file management system.

The input transaction processing of a DBMS enables the user to define input formats, validate input elements, convert data to internal formats, and then store the data in the file. Verification methods include range verification, internal comparison, and input sequence checking. Logical maintenance provides for preprogrammed file maintenance, and interactive maintenance allows file manipulation from an on-line terminal. File reorganization routines permit the reorganization, merging, and restructuring of files in preparation for data output (Sayers, 1971).

Data retrieval is an important aspect of DBMSs. Simple query languages provide flexibility to control the data retrieval. Instructions usually consist of a data base field, a logical operator such as "equal to," "greater than," or "less than," and an operand. The operand may be a constant, another data field, the result of a nested instruction, or an arithmetic instruction. Instructions may be connected using logical conjunctions such as "and," "or," and "not" (Sayers, 1971).

Queries may search single files, multiple files, or chain to secondary files based on results of the first file search. Queries may be made in a batch or interactive mode.

In addition to standard requests, such as for directory listings and input transaction listings, user-formatted reports may be produced using instructions similar to those used for data retrieval. The user may be required to write a program to produce the required program generator. Functions controlled by report programs include labeling, data formatting, and top-of-form control (Sayers, 1971).

2.4.5 Application Software

The second major software grouping is application software. Whereas systems software is concerned with the internal operations of the computer system, application software refers to programs for activities external to computer operations and specific to jobs accomplished by the computer system.

In the past, many users used their own programming staff to develop application software. An internal programming group was responsible for designing, developing, supporting, modifying, and upgrading all software specific to one site. As computers have become less expensive and thus available to more and more users, applications software packages have been developed. Software packages are standardized and tailored to the growing number of organizations needing DBMS, retail sales, inventory, manufacturing, payroll, accounting, and other programs. Smaller and first-time users cannot afford the considerable software investment needed to use relatively inexpensive hardware. Software packages can be purchased as is or customized for the user, or turnkey systems can be obtained. Turnkey vendors provide a complete computer system including hardware, system and application software, installation, testing, documentation, and training.

The lower costs of applications packages are offset by some disadvantages, some of which are the following (Kenney, 1978):

- Standardized or modified packages may not fit a user's operations.
- Software is available for only a limited number of applications.
- Once packaged software is purchased, the user must depend on its supplier for modifications and upgrades.
- Packaged software can not be easily changed.

2.4.6 Software Reliability and Maintenance

Software reliability can be determined only after a program is in use. Its reliability can be measured by taking the number and degree of errors found and by rating user satisfaction. Neither the numeric nor subjective method is adequate, however.

As with hardware, software has a high failure rate in early life and again as it ages. Early programming errors are found and repaired as the program is first developed and tested. Later, the software "fails" because it is obsolete, outmoded by newer software techniques, new hardware requirements, and a changed applications environment.

Software errors are eliminated during software development by running the program in different test environments. With each test, the more obvious errors are found but soon diminishing returns require higher cost tests to locate the remaining errors. Programs designed to operate in a limited environment can be tested easily under most expected conditions and therefore have a very high degree of reliability. The programs that must operate in multiple environments or in a very general environment cannot be tested for all error possibilities. The reliability of general programs, such as systems software, therefore is lower.

Typically, a finished software product has one error per 100 lines of program and 1,000 errors per system software release. Of these errors, estimates are that half are not programming errors but are errors in understanding the applications requirements (Champine, 1978).

The production of software remains an art. Little has been accomplished to improve development methods, production control, or quality control. Some of the methods used to improve software reliability and programmer productivity are use of programmer teams, structured programming, top-down design, and design review walkthroughs.

The hardware vendor usually provides and maintains the system software. Software is licensed to the user at a one-time cost and maintenance is usually provided at no charge for a specified period. After that time, a user may buy continued software support. This may include the correction of software problems and provision of future releases of updated compilers and updated operating systems. If a user does not have the most recent software, problems may occur when support is needed.

Applications packages also carry a guarantee of a specific duration. After that period, however, obtaining software support may be difficult. If the user has made any modifications to the program, the vendor can later argue that the origin of the errors is uncertain.

2.4.7 Software Documentation

Software documentation provides an explanation of software design that can be invaluable when changes must be made in the future, especially when the original programmers are no longer available. Documentation comprises both comments within the program and separate documents.

Documentation should include an overall functional design of the system, specifications for data files and formats, input and output formats, and a detailed description of each module of the program and how it interacts with other parts of the program (Martin, 1965). Software documentation should be a basic reference instructing the user in how to use the software.

2.5 NETWORK ORGANIZATION

2.5.1 Centralization

Typically, the high cost of computers has forced the centralization of computing resources to take advantage of the economies of scale of a centralized facility provides. In a centralized computer system, processing and data storage are at one central location. Among the many advantages of having centralized data processing and storage are (Champine, 1977; Kenney, 1978):

- Economies of scale in hardware, programming, and operations cost.
- Demand smoothing.
- Centralized implementation.
- Unified control, coordination, cost accounting, and security.
- Hardware/software compatibility.
- Easy update, retrieval, and control of a centralized data base.
- Coordinated system and software development.
- Large, well-rounded central staff with greater depth of abilities.
- No redundant data files; maximum use of storage.

Despite these advantages, large central computers are a problem to some organizations. With a single computer system, if any one link in the system fails without a backup, the whole system fails. As centralized systems grow, more components are involved, thereby increasing failure rates.

If an organization is growing, the entire central system must be replaced periodically by a larger system. Because of their size, centralized systems can only be enlarged in very large and expensive increments. The difficulties of moving all application software and peripherals to a new system can be considerable (Scherr, 1978).

Moreover, in many organizations, centralized data processing does not correspond with the management style. Decentralized management practices require more local control of resources and an accounting system focused on regional and divisional profit centers (Champine, 1977). Centralized computer operations do not fit well in this management approach because the central facility usually falls outside the normal corporate organization. High overhead costs and low responsiveness to line managers have plagued many companies (Scherr, 1978). For many organizations, centralization results in a detrimental complexity of scale (Becker, 1978).

2.5.2 Decentralization

A popular trend in recent years has been decentralization of computer resources because the traditional hardware economies of scale are no longer being realized. Processing and storage performance-to-cost ratios are 10 times those of 10 years ago (Kenney, 1978). During the same time, however, communications costs have decreased at only half that rate (Champine, 1977; Scherr, 1978). For organizations with many geographically dispersed users, communication to these remote sites has become a major cost consideration. Some organizations have found that decentralized processing can be cost-effective in reducing communications requirements (Champine, 1977).

For organizations where computer availability is important, decentralized processing can provide better reliability. The design and effectiveness of a distributed processing system depend on a proper distribution of data. If most critical or often-accessed data are site specific, they can usually be physically distributed among independent processing sites.

Some advantages resulting from a distribution of computing resources between remote sites are:

- The system remains available through central system failure and need not totally rely on data communications hardware.
- Complexity of scale is reduced by distributing processing and files to remote sites.
- Better price/performance for some specialized applications is realized because of reduced overhead.
- Application development, implementation, and expansion are independent.
- Interactive response time is better.
- Greater control and involvement of users are possible.
- It permits decentralization of functions in a structure parallel to a company's management organization but maintains central control.
- Growth and upgrades can occur in much smaller physical and cost increments.
- Information is accessible both centrally and locally in efficient site-specific file structures. Independent failures do not affect data base access, and the communications volume is reduced.

Disadvantages of decentralized systems include:

- Hardware, software, and personnel may be duplicated.
- Control of system development, standards, and expense is difficult.
- With an increased number of components in a fragmented system, more failures as a whole may occur.
- Efficiencies of distributed processing are lost when multiple sites often access the same data.
- If the data are replicated at different points to speed accessing, maintaining the accuracy of all sets of data is difficult.
- Development and training are difficult.
- Hardware maintenance is costly.
- Security risks are increased.

Three situations can be regarded as typifying distributed processing (Champine, 1977):

(1) Large volumes of input and data-accessing transactions occur at geographically distributed points. Fast response times are needed. Communication to the central or other sites is infrequent and of low volume.

(2) Large volumes of data are generated at a central location. Data can be changed either centrally or at one of many remote sites. Quick access to the information is required.

(3) Remote locations generate and maintain their own large data sets. These remote data bases must be quickly accessible for inquiry and change. The need for one node to require information of another is infrequent.

In a decentralized organization, the cost of growth is site specific. Thus, the cost need not be carried as overhead by all the users. The decision to have some distributed processing need not limit the design to distributed processing, however. A combination of processing structures may best fit the organizational structure and management policies.

2.5.3 Functional Distribution

A simple centralized processing system can be viewed as an I/O module communicating through an operating system to an application program and processor. The application program and processor in turn communicate through an operating system to a data storage module. This simple system can be divided for distribution at three different points (Scherr, 1978):

(1) Between the I/O module and the application program.

(2) Between the application program and the data storage module.

(3) Within the application program.

If the system is split between the I/O module and program, the independent processor is called a front-end processor. This type of processor exclusively handles I/O processing, such as code conversion, message queuing, polling of remote terminals, and error recovery (Couger & McFadden, 1975). This split provides an efficient distribution of processing when the I/O load is great. Examples are a network of terminals and terminals requiring a significant amount of screen formatting (Scherr, 1978).

When processing is split between the data storage module and the program, a back-end processor is created. Such a split provides a central access monitor for single or multiple storage devices.

A back-end processor can also be used as an independent data base processor.

In the third division (a centralized computer system within the application program), the structure of the application program determines how and whether the processing can be completed at different nodes. If processing can be split into tasks of equal importance and similar function, the distribution is termed "horizontal" (McGlynn, 1978). An example would be the geographic distribution of similar tasks, such as accounting, entered and processed at the regional headquarters of a nationwide organization.

A vertical or hierarchical distribution is created when processing is split into multiple components of differing sizes and functions, and a rigid multilevel master-to-slave structure is created (McGlynn, 1978).

This chapter discusses the gathering, processing, storage, transmission, and retrieval of inventory information in classification yards. A description of railroad operations is presented first, and then the computer functions are described.

3.1 RAILROAD OPERATIONAL DESCRIPTION

As a railcar moves through the classification yard, information on the car is updated to reflect its present position and status so that an accurate and up-to-date inventory can be maintained.

3.1.1 Manual PICL

Two major types of yard inventory systems are used to monitor car location and track status in the yard: manual and computer inventory systems. In the manual PICL (perpetual-inventory and car-location) system, punched cards are produced representing each car. A rack of pigeonholes is used to represent each track in the yard. The yard office clerk manually sorts the waybills and punched cards by referring to the switch list. The selected pigeonhole corresponds to the track on which the individual car was switched. In this system, the physical track-to-track car movement is followed by moving the waybills and punched cards from pigeonhole to pigeonhole. Yard personnel can determine the cars on any track by looking at the punched cards in the corresponding pigeonhole. This manual system is also referred to as a card-PICL system.

A manual PICL system should provide centralized and current information on the location of cars. Unfortunately, centralization requires considerable communication between personnel requiring the information. Providing current information is difficult because of the use of antiquated data processing equipment, the difficulty of maintaining the card inventory, difficulties in information reporting and retrieval, and the delays in providing switch lists at remote locations. Human errors also frequently hamper the system.

3.1.2 Disk-PICL

An alternative is the use of a disk-based computer inventory system. The physical movement of cars is paralleled by the movement of car records among computer files. Instant and direct access to these files is possible via computer terminals located at sites throughout the yard. Among the advantages of the disk-PICLs are:

- Faster and more efficient data storage and inventory processing.
- Fast, efficient reporting and retrieval of information using CRT terminals and direct links to other computers.
- Increased inventory control by giving respective departments complete inventory maintenance responsibility.
- Reduced manpower needs by reducing and eliminating manual tasks.

A track-standing inventory is usually kept for receiving, classification, and departure tracks. The inventory includes the cars on each track and their order on the track as well as the minimum car information. A semi-track-standing inventory keeps a record of the cars in each track but not their order; an MIS computer keeps only this level of inventory detail. Unless more detailed records of car position are kept as part of a yard inventory subsystem, this subsystem can be part of either the MIS or PC system. Exhibit 3-1 is an inventory of arriving cars and cars in the receiving classification, and departure yards by block number in Santa Fe's Barstow Yard. Within each area of the classification yard, information is kept on that area's operations.

The rest of this discussion is structured according to the movement of cars through each area.

3.1.3 Receiving Yard

Before a car arrives in the classification yard, an advance consist of the arriving train is transmitted from the previous yard or the railroad's central office by telephone, teletype, or a computer data link. The data received are stored in a "train-due file." The advance consist contains the car ID number and lading, origin, destination, and routing information. This information stays in MIS files throughout the car's stay in the classification yard and is included in the outbound consist when the car leaves the yard.

In addition to individual data on the car, the advance consist may provide the train ID number; the number of loaded and empty cars; tons carried; length of the train; lead unit number, total units, and helper units; and estimated time of arrival. After receiving the advance consist, the yardmaster may assign the receiving track and the route through the yard. Once a track is assigned, the receiving yard display indicates that the track is reserved.

When the train arrives, ID numbers are recorded using videotape, audiotape, automatic car identification (ACI) scanners, or pencil and paper.

EXHIBIT 3-1 SAMPLE CAR INVENTORY BY BLOCK NUMBER

BLOCK VOLUME 79338-1349

BLOCK #	DESC		DUE	R-YD	B-YD	D-YD	OTHER	TOTAL
1	WAYCARS		5	11	13	13	7	49
2	BOWL TRK 02		0	2	24	1	96	123
6	GLENDALE		0	18	6	31	2	57
7	BARSTOW (HOLDS & MISC)		27	87	66	3	421	604
9	MODESTO-EMPIRE JCT		8	22	26	52	2	110
10	BAKERSFIELD & AREA	GT10	1	14	15	64	3	97
11	PHOENIX	GT11	1	8	17	34	0	60
12	BORON LOCAL	GT12	0	9	8	11	0	28
13	ALBQ WINSLOW & EAST		0	15	13	0	1	29
14	CALWA & AREA	GT14	3	17	14	31	4	69
15	CLOVIS		0	5	95	1	69	170
19	HOBART		14	18	2	71	2	107
20	BARSTOW UP		7	7	21	0	2	37
21	RICHMOND AREA & TOFC		16	50	33	17	4	120
22	KANSAS CITY & BEYOND		13	57	52	42	2	166
24	STOCKTON-WP		3	14	27	1	0	45
25	STOCKTON-WPBN		1	5	18	1	0	25
26	HARBOR SHORTS	GT26	15	24	14	41	5	99
29	TORRANCE		0	14	13	26	1	54
30	SAN DIEGO & AREA		2	11	4	1	48	66
32	STOCKTON & AREA SSP		5	6	14	31	0	56
34	SLSF TULSA	GT38	0	5	2	0	0	7
35	3RD DISTRICT SHORTS	GT35	0	12	14	21	41	88
36	1ST DIST SHORTS	GT3	0	5	0	0	11	16
37	PICO RIVERA		41	6	3	20	1	71
38	SLS2 MEMPHIS&BEYOND	GT38	1	19	11	0	0	31
39	KAISER & 2ND DIST SHTS		5	15	8	1	33	62
41	SAN BERNARDINO	GT41	4	8	1	1	30	44
42	PARKER LOCAL SHORTS	GT42	0	15	3	12	0	30
44	SANTA ANA		0	4	2	11	16	33
46	SNYDER RSP	GT16	4	0	1	9	2	16
49	MOJAVE DIST SHORTS		0	6	2	0	0	8
51	BARSTOW STGE YARD		0	0	4	0	45	49
52	BARSTOW TO SALTUS SHORTS	GT33	0	0	2	0	2	4
55	ALBQ & NS DIST SHORTS	GT33	0	3	2	0	21	26
57	ABILENE SET OUT	GT16	0	0	0	4	0	4
58	FORT WORTH	GT16	2	8	1	7	1	19
59	NEW ORLEANS	GT16	0	0	0	3	0	3
60	BROWNWOOD	GT33	2	8	4	26	0	40
61	HOUSTON	GT33	0	2	0	5	0	7
62	LUBBOCK	GT33	0	0	0	1	0	1
63	DALLAS	GT33	1	2	0	10	0	13
64	FORT WORTH	GT33	0	6	3	17	0	26
66	FULLERTON ORANGE AREA		0	4	2	18	1	25
67	FULLERTON ORANGE AREA		1	3	7	16	0	27
68	FULLERTON ORANGE AREA		0	0	2	6	0	8
69	FULLERTON ORANGE AREA		0	10	1	6	0	17
71	BELEN	GT23	0	4	4	0	0	8
72	AMARILLO	GT23	0	2	4	0	0	6
73	EMPORIA	GT23	0	2	8	4	0	14
74	WELLINGTON	GT23	0	0	1	4	0	5
75	KIOWA	GT23	0	0	4	10	1	15
77	LAMIRADA		7	9	7	37	0	60
81	ALBUQUERQUE		0	15	6	0	0	21
82	TRINIDAD CS		0	0	7	0	0	7
83	PUEBLO & BEYOND		0	8	8	0	0	16
84	LAJUNTA		0	0	1	0	0	1
90	VALLEY PARK MO - 2290SLSF		0	7	0	0	0	7
91	OCEANSIDE		0	3	2	0	7	12
99	*UNASSIGNED*		0	0	1	5	0	6
101	BARSTOW NEW YARDS		0	11	0	0	10	21
109	*UNASSIGNED*		25	28	11	39	4	107
115	ANTIOCH		0	1	8	2	0	11
116	PITTSBURG		2	0	1	3	0	6
119	LOSANGELES SP		10	12	0	19	0	41
125	*INVALID STATION NUMBER*		0	1	6	2	69	78
TOTAL			226	658	549	791	944	3042

Courtesy of Barstow Yard, The Atchison, Topeka and Santa Fe Railway Company.

This information is checked against the inventory of the arriving train. Corrections to the inventory are made, and arrival time and track location are added to each car's record. Individual car inventory records are then created in the inventory file. If the arriving train originated locally, the yardmaster will not have received an advance consist. In this case, yard clerks must enter car information manually when reviewing car ID numbers and when preparing the waybill from bill of lading information. An expected completion time for prehump activities may also be noted.

The car location is updated in the inventory with each movement of the car through the yard. Waybills, bills of lading, and other paperwork are given to the yard or arrivals office clerk. The status of each newly arrived car changes on completion of mechanical and inspection work, the completion of incoming paperwork, and the generation of the switch list.

A switch list is a list of the cars to be humped, in track-standing order, that designates classification track destination. Switch list is also used as a general term for cut slip, hump list, and classification guide, which may have more specific definitions in some classification yards. Switch lists may be generated manually by the yardmaster or automatically by the MIS computer; Exhibit 3-2 is a sample of a computer-generated switch list. A classification number or block number is assigned to and remains with each car throughout its stay in the yard (Allman, 1968). Given the receiving track to be pulled and the track assignment of each block, a list of track destinations for each car is made. In Southern Pacific's West Colton Yard, the terminal control computer (TCC, a yard MIS computer) automatically sends the crest control computer (CCC, a PC computer) the switch list when requested. The CCC then controls all switching and retarding automatically. In Southern's Inman Yard, the engine foreman uses a printed switch list to manually switch cuts. (A cut is one or more consecutive cars going to the same classification track. A cut list designates the order of cars to be humped and how they are to be grouped as cuts.)

3.1.4 Classification Track

A car's location in the classification yard is changed in the inventory according to the switch list. The inventory change can be made in real time (as the car is humped) via the PC computer, or the change can be posted from the completed switch list. Any misswitched cars must be detected and the inventory adjusted. A sophisticated PC computer may measure the weight and length of a vehicle and each track's distance-to-couple measurement and add this information to the inventory record.

Misswitched cars may be detected by visual inspection or by the PC computer. Another method is to count with wheel detectors the number of cars entering and leaving classification tracks.

EXHIBIT 3-2 SAMPLE COMPUTER-GENERATED SWITCH LIST

SWITCH LIST 1 JOB-ID CURT DATE 15-OCT-79 TIME 01:41 PM

YARD 1 TRACK 1 DIRECTION S LENGTH 650 CUT-ID

SEQ	INITNUMBER	LK	YBLK	TRK	SPCD	CNTNT	*NEXT*SYS*DEST*	CONSIGNEE	TRN-ID	TBLK	SEQ
1	YATS002500	LD	DUPO	10		BEER	X 344 28 SSW	NLITROCK	OSNL27	SSW	1
2	YATS002600	LB	DUPO	10		BEER	X 344 28 SSW	NLITROCK	OSNL27	SSW	2
3	YATS002700	LD	DUPO	10		BEER	X 344 28 SSW	NLITROCK	OSNL27	SSW	3
4	YATS002800	LD	DUPO	10		BEER	X 344 28 SSW	NLITROCK	OSNL27	SSW	4
5	RICH001000		CM	999			CM			#0	5
6	QWER000001	ED	KCMO	8							6
7	RICH000022		CM	999							7
8	PT 201180	LB	CM	999		BLGPA	AX340 CM	CELOTEX			8
9	LN 023972	EB	DA2	999	??			EVANSVILL			9
10	RICH014000		CM	999			CM				10
11	RICH021000		CM	999			CM				11
12	RICH013000		CM	999			CM				12
13	YATS005800	EN	HC	10			MX001 HC NW	ABT			13

END OF LIST

Courtesy of Missouri Pacific Railroad

When the physical inventory count differs from the expected count, the inventory is adjusted. The inventory system must also keep track of individual exceptions that occur at the hump. Examples include (Graziani, 1977):

- Overweight cars
- Side-shift cars
- Bad-order cars
- Hold cars
- Rehump cars.

Inventory is not exclusively an MIS application. A parallel inventory may be kept in the PC computer for redundancy. The PC computer usually stores only the receiving yard inventory or an inventory of the four or five cuts waiting to be humped and the cuts just humped that have not been updated in the MIS inventory.

Changes in cars' locations on classification tracks are usually posted to inventory from the completed work order or switch list.

3.1.5 Industry Tracks

The inventory system of a classification yard may also be used to monitor cars on industry tracks in the area. This inventory includes information on the location of the cars, their loading/unloading status, the shipper's name, and the date and time of arrival. A bulk inventory is kept for industry tracks. The location of cars is kept only by geographic areas, because only a small number of cars are at specific sites. The MIS computer can use this information to compute length of stay for demurrage charges and alert the railroad of extended delays. This can be very important because demurrage charges do not cover wasted car utilization (Sargent, 1974).

3.1.6 Repair Tracks and Yard Vehicles

The yard inventory system can be used to monitor cars in repair yards and permanent yard vehicles such as switch engines. The inventory of repair tracks, engine tracks, and work equipment tracks is usually a semi-track-standing inventory.

If a car is found to be damaged when it is inspected in the receiving yard, the switch list is changed automatically or manually at humping so that it can reflect the car's location on a bad-order track or in the repair yard.

The inventory of yard vehicles need not accurately reflect their current location because they are continuously being moved. Nonetheless, it is important to know of any out-of-service vehicles and to store information regarding the scheduling of preventive maintenance.

3.1.7 Departure Yards

Outbound trains are assembled in the departure yard according to a makeup list, which is compiled by the yardmaster manually or with computer assistance. As each classification of cars is pulled from the classification tracks, inventory records are updated manually or through the MIS computer. Cars are grouped in inventory according to their train number and scheduled departure time. Because each car record contains a weight figure, a total weight for the train can be calculated to estimate the number of engines needed. Car IDs and waybills are compared with the outbound consist, and the train is mechanically readied. The outbound consist is sent by telephone, teletype, or a computer data link to the next yard or to a central-office or system-wide MIS computer; it then becomes an advance inbound consist.

The MIS computer of a classification yard can also be used in the preparation of outbound trains. Such a system requires that the number of outbound trains be entered into the computer. Using inventory destination information, the computer can display the cars that are appropriate for a particular train. Each car is displayed with its location, destination block code, loaded or empty status, weight, length, and special handling requirements (McClellan, 1977).

Canadian Pacific's yard MIS system (YARDS) presents a yard scan track map on a CRT screen to display car locations in the yard. The "map" gives the track number and total number of cars for each track and displays a single-character outbound train code in the location of each car. An example is as follows:

Track No.	Car Total	Train Codes
14	11	PPLE...E31.466
15	15	22K33...466.7JJJ88

The yard scan given can be the actual or projected map after switching. The tonnage report of Potomac Yard (RF&P) displays the total number and weight of cars on each track of a departing train (Exhibit 3-3).

EXHIBIT 3-3 SAMPLE TONNAGE REPORT

```
TN
ENTER TRAIN SYMBOLR109
AFTER TRACK   1 RW6
  3 REGULAR 28 CARS IN TRACK 28    210 TONS
 14 REGULAR 21 CARS IN TRACK 21    830 TONS
  7 REGULAR 26 CARS IN TRACK 26    355 TONS
TRAIN 109     24 TOTAL CARS   1395 TOTAL TONS
ENTER REQUEST
```

Courtesy of Potomac Yard; Richmond, Fredericksburg, and Potomac Railway Company.

The MIS computer can also be used to select the best departure track and makeup engine (Petracek et al., 1976). By adding the weight of all the assigned cars and comparing it with the power of the assigned cars and available engines, an optimal solution can be found. Taken one step further, the computer can be used to issue crew assignments and track makeup instructions.

3.1.8 Information Retrieval

In addition to keeping an inventory by ID number, advanced disk-PICLs can provide other inventory information, including:

- Block summaries (i.e., block count) by area, zone or track (see Exhibit 3-1).
- Yard summary--number of cars on each track (see Exhibit 3-4).
- Track-standing inventory (see Exhibit 3-5).
- Track overflow report, detailing the length of cars on each track and the length of the track (see Exhibit 3-6).
- Track status report (e.g., track assignment, spiking tracks, overflow, maintenance, out of service).
- General-purpose inquiry reports--e.g., Where is car "X"? (Exhibit 3-7); where are all cars of type "Y" (Exhibit 3-8) or class/block "Z"? (Exhibit 3-4).
- Car characteristics--weight, length, list by ID number.
- Individual car inquiry by ID number (Exhibit 3-9).

3.1.9 Management Evaluation Reports

Management evaluation reports can summarize the current yard status in more detail than is available from on-line inquiry. Management reports can also summarize past activities in the yard. Among the types of management evaluation reports produced are (Wert & McGlumphy, 1968):

- Daily activity summaries--hump rate, number of inbound and outbound trains Exhibit 3-10).
- Exceptions to the work plan.
- Number of trains and cars classified (Exhibits 3-11 and 3-12).
- Number of misswitched cars.
- Number of damaged incoming cars and shop reports (Exhibit 3-13).
- Summaries of work completed, car movements, engine movements.
- Summaries of car location by track and yard.

EXHIBIT 3-4 SAMPLE FOUR-HOUR REPORT

H4P
FOUR HOUR REPORT TO DESK 223 06/11/80
AFTER TRACK 2 120 1 PW6

TRK	CLASSIFICATION	LOADS	MTYS	CARS	
6	WASHINGTON	3	6	9	
24	OAK ISLAND	13	3	16	
26	CROXTON	11	0	11	
35	SELKIRK	111	6	117	
21	FRANKFORD JCT	10	0	10	
22	CAMDEN/PAVONI	11	0	11	
34	ABRAMS	1	5	6	
29	ALLENTOWN	29	13	42	
32	ENOLA	5	5	10	
27	READING	6	11	17	
28	CONWAY	8	2	10	
40	METUCHEN	0	0	0	
19	PHILA PB	0	0	0	
38	KEARNY PB	2	6	8	
30	BALTIMORE	21	11	32	
23	MORRISVILLE	14	1	15	
33	EDGE MOOR	15	1	16	
TOTAL	LOADS 260	EMPTIES	70	CARS	330
42	PENNSYLVANIA	4	0	4	
43	CANADIAN	8	1	9	
45	D&H OTHER	2	4	6	
39	D&H MECHV. BM	13	1	14	
TOTAL	LOADS 27	EMPTIES	6	CARS	33
48	76 WASHINGTON	11	0	11	
49	74 BRUNSWICK	19	8	27	
37	70 CUMBERLAND	18	34	52	
12	80C CUR. BAY	1	0	1	
13	80LM LOC. PT.	4	0	4	
11	92AB ABRAMS	2	1	3	
10	92 PARK JCT	2	0	2	
9	865 ELIZ'PORT	0	0	0	
8	90 PHILA.	0	1	1	
14	82 BAYVIEW	2	1	3	
16	92M M'VILLE	5	0	5	
17	86 WILSMERE	2	0	2	
18	92 D&H	6	1	7	
TOTAL	LOADS 72	EMPTIES	46	CARS	118
20	LOCALS	4	0	4	
25	FREDERICKSBUR	0	0	0	
23	RICHMOND-ACCA	8	9	17	
9	FLORIDA PTS	35	5	40	
13	ROCKY MOUNT	6	12	18	
0	HAMLET	18	18	36	
15	FLORENCE	6	1	7	
17	SAVANNAH	5	6	11	
TOTAL	LOADS 82	EMPTIES	51	CARS	133
6	ALEXANDRIA	2	2	4	
4	MACON	4	13	17	
10	MILLER	2	0	2	
8	BIRMINGHAM	1	0	1	
0	ATLANTA	4	1	5	
11	LINWOOD	20	29	49	
1	JACKSONVIL PB	0	0	0	
2	TOFC MIXED	0	0	0	
3	DANVILLE	1	6	7	
12	GREENSBORO	1	7	8	
19	LOCALS	7	1	8	
TOTAL	LOADS 42	EMPTIES	59	CARS	101
99	C&O	38	3	41	

ENTER REQUEST

Courtesy of Potomac Yard; Richmond, Fredericksburg, and Potomac Railway Company.

In a second type of management evaluation report, operational statistics and standards are used to evaluate the efficiency of the yard. Examples include:

- Track utilization.

EXHIBIT 3-5 SAMPLE TRACK INVENTORY INQUIRY

```
T2
ENTER TRACK THEN
ENTER S FOR SOUTH N FOR NORTH 39S
AFTER TRACK    1 RW6
TRACK 39 SOUTHBOUND CLASS YARD
 6 LOADS    5 EMPTIES   11 CARS    671 TONS
POS INIT NUMBER    OT   AT
  1 RFP     3113 E 29   39              MTY     DAVINDUST
  2 RFP     3147 E 39   39              MTY     DAVINDUST   9585
  3 NW     56760 L 39   39              CIGARE  RJREYTOBA   9573
  4 NW     56776 L 39   39              CIGARE  RJRTOBC     9573
  5 RFP     2430 E 23   39              EMPTY
  6 RFP     2417 E 23   39              EMPTY
  7 RFP     2447 E 23   39              EMPTY
  8 SOU    32255 L 39   39              BRICK   POTVALBRI   9591
  9 SOU    32105 L 39   39              BRICK   POTVALBRI   9591
 10 SOU    32138 L 39   39              BRICK   POTVALBRI   9591
 11 SOU   530299 L 39   39              BRICK   POTVALBRI   9591
ENTER REQUEST
```

Courtesy of Potomac Yard; Richmond, Fredericksburg, and Potomac Railway Company.

EXHIBIT 3-6 SAMPLE TRACK INQUIRY REPORT

```
                              TRACK INQUIRY

TRACK  1   YARD 1  LISTED S-N  01:41 PM 15-OCT-79
TOT CARS: 13  LOADS: 5  EMPTIES: 8  CAR FT: 635  TONS: 586
TRK FT: 3133  STATUS: 1  SWITCH LIST(S)
SEQ INITNUMBER LKND YBLK SPCD CNTNT *NEXT*SYS*DES* IB-TRN IB-DA TRN-ID IO-ID ST
  1 YATS002500 LD4U DUPO      BEER  X 344   28 SSW         82607 OSNL27 OSNL2701
  2 YATS002600 LB4D DUPO      BEER  X 344   28 SSW         82607 OSNL27 OSNL2701
  3 YATS002700 LD4U DUPO      BEER  X 344   28 SSW         82607 OSNL27 OSNL2701
  4 YATS002800 LD4U DUPO      BEER  X 344   28 SSW         82607 OSNL27 OSNL2701
  5*RICH001000        CM                 CM                22011                01
  6 QWER000001 ED4U KCMO                                   72008                01
  7 RICH000022                                             31314                01
  8 PT  201180 LB5H   CM      BLGPA AX340CM                62503                01
  9 LN  023972 EB   DA2  ??                         RICH29 82215                01
 10*RICH014000        CM                 CM                22011                01
 11*RICH021000        CM                 CM                22011                01
 12*RICH013000        CM                 CM                22011                01
 13 YATS005800 EN4A   HC            MX001HC NW      SLSF   82614                01

END OF LIST
```

Courtesy of Missouri Pacific Railroad.

EXHIBIT 3-7 SAMPLE INDIVIDUAL CAR INQUIRY

```
W1 SOU   15206
INVALID ENTRY
ENTER REQUEST

W1 SOU  152063
SOU  152063         11 11             SBD CLASS YD
ARRIVED    154      0130 0608    03 CLASSIFIED
INTERCHANGE         0127 0608       REPORTED      SOU  SOU
CLASSIFIED          0110 0611 80
L F  045 015 HOSE    3522991 NORWICH    ON RAYMUR
724-SOU -POTYD-SOU -BUFF -CN  -CN
91 X       0605 005544  01912 BARMAC     NC IRRUSA

15BE  2230 0608       03
SHOP  1600 0609       04
ETOA  1410 0610       11
ENTER REQUEST
```

Courtesy of Potomac Yard; Richmond, Fredericksburg, and Potomac Railway Company.

EXHIBIT 3-8 SAMPLE DANGEROUS CARS REPORT

```
DX
DANGEROUS CARS LOCATED IN NORTHBOUND CLASSYARD
                AFTER TRACK 05 156
INIT NUMBER   OT AT
UTLX  50040 D  3  3 DANGEROUS
ACFX  24622 D  3  3 DANGEROUS
UTLX  96908 D 29  3 DANGEROUS
FMLX  14018 D 12 12 DANGEROUS
UTLX  74162 D 13 13 DANGEROUS
ACFX  88240 D 14 14 DANGEROUS
GATX  50801 D 32 32 DANGEROUS
ACFX  13946 D 33 33 DANGEROUS
ACFX  23845 D 37 37 DANGEROUS
NAHX  52411 D 43 43 DANGEROUS
DANGEROUS CARS LOCATED IN SOUTHBOUND CLASSYARD
                AFTER TRACK 01 RW6
INIT NUMBER   OT AT
GATX  44940 D  3  3 DANGEROUS
ACDX   9803 D  9 11 DANGEROUS
ACDX   9669 D 11 11 DANGEROUS
GCX  417133 D 13 13 DANGEROUS
ACFX  14504 D 13 13 DANGEROUS
NATX  17398 D 13 13 DANGEROUS
NATX  17394 D 13 13 DANGEROUS
GATX  44919 D 14 14 DANGEROUS
UTLX  84863 D 36 36 DANGEROUS
TGAX 141521 D 37 37 DANGEROUS
ENTER REQUEST
```

Courtesy of Potomac Yard; Richmond, Fredericksburg, and Potomac Railway Company.

EXHIBIT 3-9 SAMPLE CAR INQUIRY

```
            YATS CAR INQUIRY  01:42 PM  15-OCT-79
INIT NUMB L/E KND TON FT  CONT   ***NEXT*SYS*DEST***  **DEST*INFO**
YATS002500 L  D4U  69 45 BEER    X 344DUPO    28 SSW   NLITROCK AR
YARD BLOCK:DUPO TRAIN ID:OSNL27  TRAIN BLOCK:SSW
CURRENT LOCATION:  IN YARD   1 ON TRACK 1 AT 09:14 AM 11-OCT-79
CAR STATUS:  SWL 01 SET-0--OSNL27
MV YD TRCK MOVMNT TIME & DATE     INB/OUTBND    TRAIN     TIME      DATE
                                  ARR YARD 01             07:00 AM 26-AUG-79
                                  LAST UPDATE:  08:20 AM 12-OCT-79

            YATS CAR INQUIRY  01:42 PM  15-OCT-79
INIT NUMB L/E KND TON FT  CONT   ***NEXT*SYS*DEST***  **DEST*INFO**
YATS002600 L  B4U  69 45 BEER    X 344DUPO    28 SSW   NLITROCK AR
YARD BLOCK:DUPO TRAIN ID:OSNL27  TRAIN BLOCK:SSW
CURRENT LOCATION:  IN YARD   1 ON TRACK 1 AT 09:14 AM 11-OCT-79
CAR STATUS:  SWL 01 SET-0-OSNL27
MV YD TRCK MOVMNT TIME & DATE     INB/OUTBND    TRAIN     TIME      DATE
                                  ARR YARD 01             07:00 AM 26-AUG-79
                                  LAST UPDATE:  08:20 AM 12-OCT-79

            YATS CAR INQUIRY  01:42 PM  15-OCT-79
INIT NUMB L/E KND TON FT  CONT   ***NEXT*SYS*DEST***  **DEST*INFO**
YATS002700 L  D4U  69 45 BEER    X 344DUPO    28 SSW   NLITROCK AR
YARD BLOCK:DUPO TRAIN ID:OSNL27  TRAIN BLOCK:SSW
CURRENT LOCATION:  IN YARD   1 ON TRACK 1 AT 09:14 AM 11-OCT-79
CAR STATUS:  SWL 01 SET-0--OSNL27
MV YD TRCK MOVMNT TIME & DATE     INB/OUTBND    TRAIN     TIME      DATE
                                  ARR YARD 01             07:00 AM 26-AUG-79
                                  LAST UPDATE:  08:20 AM 12-OCT-79


END OF LIST
```

Courtesy of Missouri Pacific Railroad.

EXHIBIT 3-10 SAMPLE REPORT OF TRAINS RECEIVED AND FORWARDED

REPORT OF TRAINS RECEIVED AND FORWARDED

MONTH OF JUNE 1980

	RECEIVED FROM								FORWARDED TO							
DATE	RFP	SOU	CEO	CR	BEO	DEH	TOTAL	CUM'L	RFP	SCL	CEO	CR	BEO	DEH	TOTAL	CUM'L
1	7	8	1	8	4	1	29	29	6	6	3	7	3	1	26	26
2	6	3	3	11	3	0	26	55	6	5	4	12	3	0	30	56
3	7	5	3	5	4	1	25	80	8	4	3	7	3	1	26	82
4	7	8	2	9	4	1	31	111	7	6	1	9	4	1	28	110
5	8	4	1	11	3	2	29	140	8	7	2	8	3	1	29	139
6	6	5	2	8	3	0	24	164	7	5	2	3	2	1	20	159
7	6	6	1	14	3	1	31	195	7	5	2	9	3	1	27	186
8	7	5	1	10	4	2	29	224	6	5	1	7	3	1	23	209
9	7	5	2	6	4	1	25	249	7	5	2	10	3	1	28	237
10	7	6	1	9	3	1	27	276	8	5	1	7	4	1	26	263
11	0	0	0	0	0	0	0	276	1	3	1	1	0	0	6	269

Courtesy of Potomac Yard; Richmond, Fredericksburg, and Potomac Railway Company.

EXHIBIT 3-11 SAMPLE REPORT OF CARS HANDLED

CARS HANDLED

MONTH OF MARCH 1980

	SOUTHBOUND				NORTHBOUND				TOTAL CARS RECEIVED AND FORWARDED		
	RECEIVED		FORWARDED		RECEIVED		FORWARDED		DAILY GRAND		AVERAGE CARS
DATE	DAILY	CUM'L	DAILY	CUM'L	DAILY	CUM'L	DAILY	CUM'L	TOTAL	CUM'L	PER DAY
1	1181	1181	1237	1237	1130	1130	979	979	4527	4527	4527
2	897	2078	740	1977	709	1830	716	1695	3062	7589	3794
3	818	2896	885	2862	1085	2924	726	2421	3514	11103	3701
4	997	3893	1337	4199	721	3645	1034	3455	4089	15192	379[illegible]
5	49	3942	353	4552	0	3645	88	3543	490	15682	3136

Courtesy of Potomac Yard; Richmond, Fredericksburg, and Potomac Railway Company.

EXHIBIT 3-12 SAMPLE REPORT OF DAILY INBOUND AND OUTBOUND TRAIN STATISTICS

REPORT OF CARS PROCESSED FOR 24 HOURS ENDING 1159 PM 06/09/80

INBOUND

	TRAINS	LOADS	EMPTIES	TOTAL
RFP IND	3	0	12	12
RFP ROAD	7	392	144	536
SOU RWY	5	258	51	309
C&O RWY	2	114	35	149
CR RR	6	220	326	546
B&O RR	4	84	87	171
D&H RR	1	33	67	100
TOTAL	28	1101	722	1823

OUTBOUND

	TRAINS	LOADS	EMPTIES	TOTAL
RFP IND	3	12	3	15
RFP ROAD	7	243	409	652
SOU RWY	5	155	277	432
C&O RWY	2	137	66	203
CR RR	10	616	274	890
B&O RR	3	134	66	200
D&H RR	1	72	29	101
TOTAL	31	1369	1124	2493
GRAND TOTAL	59	2470	1846	4316

Courtesy of Potomac Yard; Richmond, Fredericksburg, and Potomac Railway Company.

EXHIBIT 3-13 SAMPLE DAILY STOP REPORT

```
     NORTHBOUND DAILY CAR SHOP REPORT   AS OF   0555   6/10/80
           IN/OUT ROAD CODE   PC=1   BO=2   RFP=3   CO=4   SOU=5   SCL=7
KPO INIT  NUMBER KD L/E RD CTENTS DESTINATION  CONSIGNEE
                 SYMBOL TIME DATE          DEFECT          HRS       COMMENTS
 P  RFP     2711 B   L  51 PAPER  PEPPERELL MA BEMIS
                 158    0415  606 CARRIER IRON              97****
 N  RSLX   11021 CX  L  51 PLSTC  HAVDEGRAC MD OWEILLIND
                 154    2055  609 AIR BRAKES INOP.            9
 R  RBOX   11091 B   L  71 TEXSCR SCRANTON  PA FEIGERDON
                 190    0200  609 DOOR BULGED                27
 E  FL     17561 G   E  71 EMPTY  SPAPT     MD BETSTEEL
                 112    1915  609 CUT LEVER                  10
 W  SCL    20591 B   L  71 PPRPPD CHESTER   PA SCOPAPER
                 112    1440  607 CLASSYDSHOP               63****
 B  CO     23051 B   E  44 MTY    OWEMILLS  MD MARPAPCUP
                 X3901  0220  608 OTHER                     51**
 L  GATX   64271 TX  E  77 EMPTY  SOLVAY    NY LOPCHEMIC
                 176    1159  604                          138**
 P  GATX   65691 TY  L  72 ACID   BALTIMORE MD DELCHEMIC
                 290    0601  606 CENTER TRAILER            95****
 P  SCL   114821 F   L  52 INGALU HALETHOP  MD KAIALUCHE
                 156    0255  607 BRAKE RIGGINS             75****
 T  CP    301041 F   E  51 MTY    BALTIMORE MD AGENT
                 154    0130  608 CLASSYDSHOP               52**
 Y  PC    369201 RX  E  77 EMPTY  RICHMOND  VA AGENT
                 112    1550  608 CARRIER IRON              38
 L  SCL   635381 B   L  71 PULPBD SYOSSET   NY GREAMECOP
                 110    0030  606 CLASSYDSHOP              101****
 T  NS      2532 B   L  51 PULPBD LEVITTOWN PA MULTIPAK
                 156    0230  608 CLASSYDSHOP               51****
 P  SCL    11262 B   L  71 FURN   HORSEHEAD NY PERFURNIT
                 176    1020  606 SILL,STEP                 91****
 T  SOO    16732 B   E  33 EMPTY  ABRAMS    PA CLEANCUT
                 120    0100  608 DOOR,SIDE                 52**
 P  ACFX   24622 TX  E  56 EMPTY  SCHENECTA NY GENELECTR
                 156    0025  609 OTHER                     29
```

Courtesy of Potomac Yard; Richmond, Fredericksburg, and Potomac Railway Company.

- Individual car throughput, time in yard versus a standard.

- Percentage of time that the hump is in use.

- Amount of time required for trimming operation.

- Switch engine utilization (cars per engine hour).

3.1.10 Interyard Information Processing

Inventory data and management information are only used within the yard and by the railway's central office (systemwide MIS) because the information pertains only to conditions within the classification yards. Consist, interchange, and financial information, on the other hand, may be sent to other rail yards and/or to another railway.

3.1.10.1 Consists. In a manual system, the consist information arrives by teletype or as a paper tape and is reproduced on a punched card for the PICL (Railway Gazette, 1965). With a computer inventory system, the information may be entered as paper tape, cards, or via a CRT display. When two computers are linked together or share a common central computer, the information can be transmitted directly.

As detailed in Section 3.1.3, consist information is stored as part of the PICL of the classification yard. When the train departs, an advance consist as well as an estimated time of arrival are sent to the destination yard of the outbound train.

3.1.10.2 Interchanges. The processing of interchange reports can be assisted by the systemwide MIS and computer-to-computer communication. Incoming receipts require as minimum data an equipment code, initial and number, and loaded or empty status indicator. For loaded cars terminating within the yard's district, the contents and industry's spot number must also be recorded. These data are usually received by telephone, teletype, telecopier, or message delivery. Other data are entered upon receipt of the waybill. When all data have been received, the MIS computer automatically reproduces statements and audits foreign statements (McClellan, 1977).

Paperwork on the outbound trains can also be interactively produced using information from the yard inventory system and waybill.

A second use of computers is the transfer of interchange information. By sending information electronically, some railways have eliminated the exchange of interchange reports printed on paper. In the case of the Southern and St. Louis-San Francisco railways, about 6,200 reports a month are now paperless. Of this number, in July 1977 Southern only found 23 exceptions (Isenberg, 1977). Illinois Central Gulf Railroad reported in 1979 that 43% of its interchange volume was by paperless transfer (Hall, 1980). Electronic interchange transfer requires fewer employees and less paper and storage, and it provides faster, more accurate interchanges with a complete audit trail.

3.2 COMPUTER FUNCTIONS

This discussion of yard MISs is limited to the functions of the yard inventory system. Figure 3-1 depicts the relationships between the yard MIS functions, and Exhibit 3-14 lists the products of those functions. In this section, the MIS software functions are described in chronological order, parallel to the physical movement of a car through the yard, and a more detailed discussion follows on the structure of inventory files and their relationship to frequency and order of inquiries and reports.

In this discussion, a number of references to the inventory files are made. Typical yard data bases divide the inventory records between three files: the car index file, in car ID order; the car record file, in random order; and the track inventory file, in track-standing order. The distribution of data between these files is determined by how often any item of information is accessed by car number or track order. Any information that is rarely accessed is stored in the car record file. That file is structured to save memory space but is more difficult and time consuming to access.

3.2.1 Overall Functional Description

3.2.1.1 Inbound processing. An advance consist may have to be translated into the standard format of the yard, especially if it has been sent from another railroad. When the train arrives, the consist is confirmed, and a receiving yard inventory is created, as follows.

(1) Receive advance consist

- Input
  - Train ID
  - Estimated time of arrival
  - Individual car information.

- Output
  - Individual car records
  - Train arrival record with estimated mix of blocks.

- Processing--The MIS computer or a front-end communications processor communicates with the sending computer.

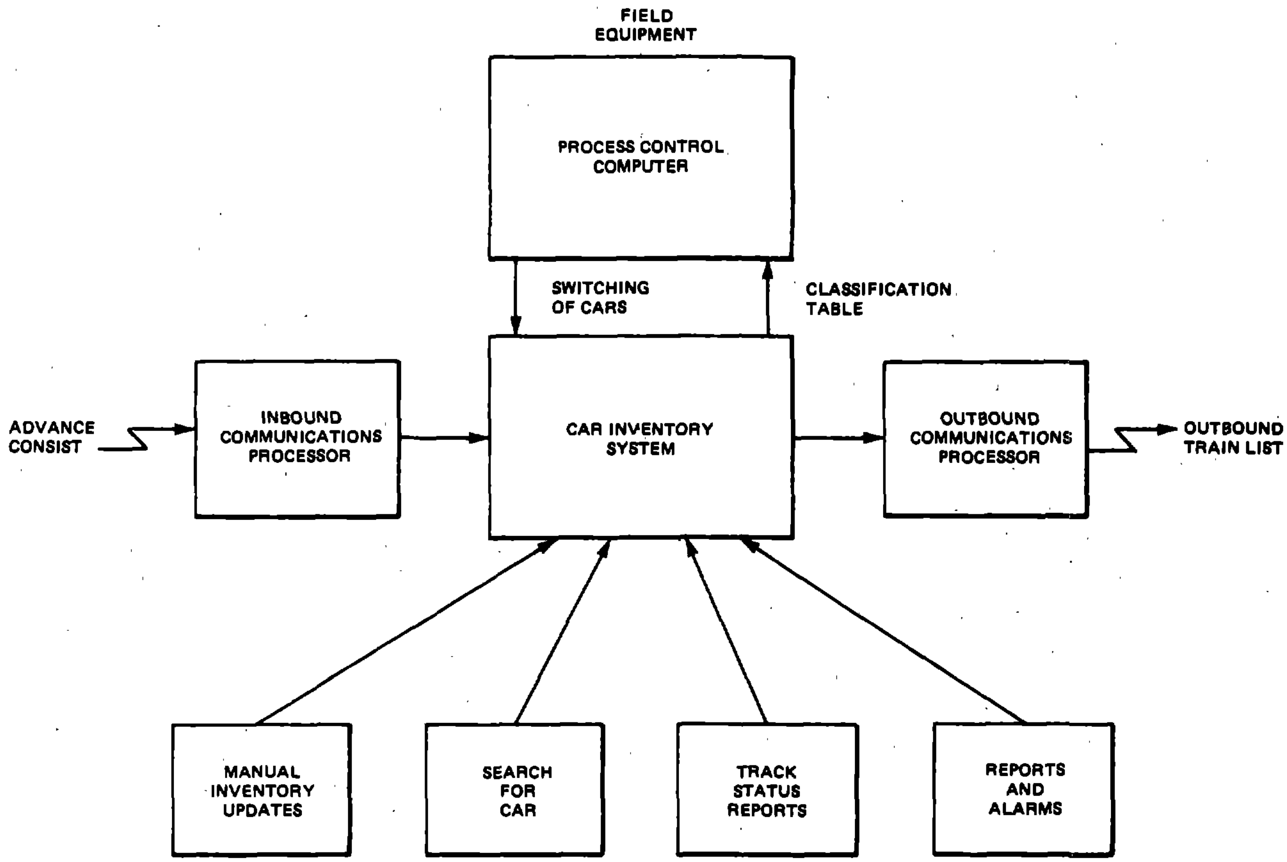


FIGURE 3-1. YARD MIS FUNCTIONS

Information on the train and individual cars is received, acknowledged, and checked for errors. The incoming car information is translated into the format of the individual car record used in the yard inventory file structure. Train arrival information and a count of car destinations are stored as a train arrival record.

(2) Empty car disposition

- Input--Arriving car record of empty car.

- Output--Confirmed destination for car.

- Processing--The destination or block of all empty cars is checked by sending an empty car disposition inquiry to the receiving railroad or to yard or the systemwide MIS computer to confirm the existing destination or to inquire whether any exists. The car record is updated when a reply is received. The central or systemwide MIS computer may also perform the function of communicating with external yards or railroads.

(3) Create receiving yard inventory

- Input
  - Record of actual car order and ID numbers.
  - Train arrival report.
  - Individual car paperwork (waybills and consists).

- Output
  - Standing-order track inventory of receiving tracks.
  - Upgraded car record.
  - Arrival record in history file.

- Processing--Verify car order in arriving train by observing videotape or television monitor, via ACI, or from inspection report. Verify and upgrade individual car records from waybills. An interactive MIS computer allows completion of this function on a CRT terminal and keyboard. On the completion of the record verification, the car records become part of the inventory car record files, and the

Detailed Car Record

- Detailed individual car records
- History of car movements in yard
- Waybill consist information.

Inventory of Cars by Position in Track

- Track-standing inventory (track occupancy from PC computer).
- Limited individual car records (weight from PC computer).
- Records of cars in shop and hold tracks
- Industry track inventory.

Arrival Processing

- Add car records to detailed car record
- Add cars to inventory of receiving yard
- Input inspection report via CRT terminal
- Update train arrival record
- Update train arrival/departure history.

Classification

- Determine train humping sequence.
- Provide automatic classification and track assignment.
- Provide automatic swing (depends on operational criteria).

Car Movement--Inventory Update

- Receive inventory updates from PC computer.
- Record moves made independent of the hump or arrival processing (via terminal or manual input).
- Produce inventory update report.

Departure Processing

- Monitor train preparation.
- Prepare consist report.
- Print outbound train sheet.
- Update inventory and train departure record.
- Update train approval/departure history.
- Move detailed car records to history.
- Prepare advance consists.

Inventory Reports

- Track-standing inventory.
- Track occupancy from inventory records.
- Inventory summary by track, total loaded/empty cars, tonnage.
- Hump sequence (humping order of inbound trains as determined by operational criteria).
- Inquiry by car ID, location, status, weight, block number, arrival/departure time.
- Track status.
- Track overflow report (distance to couple from PC computer).
- Traffic evaluation inquiry, length of stay in yard.
- Cars/tonnage per track or block or outbound train.
- Per diem by block.
- Alert tracks that need to be pulled.
- Warnings/alarms from PC computer.
- Arrival/departure summary report.
- Shop reports.

Other

- Resource scheduling
- Dynamic track assignment
- Accounting.

Communications

- To/from PC computer
- Advanced consists
- Empty car disposition inquiry
- Waybills
- Interchange reports.

inventory of the occupied receiving yard tracks is updated with the standing order of the arriving train. The arrival is recorded in a train movement history file.

(4) Upgrade car record with inspection report

- Input--Inspection report.
- Output
  - Updated car record with inspection time.
  - Reclassification of damaged cars.
- Processing--Inspection records are entered directly via a CRT terminal or as a batch input. The inspection time and results are recorded in the individual car records. Any cars that must be rerouted to repair/shop tracks or to hold tracks because of a high/wide restriction receive new corresponding classifications or block numbers.

3.2.1.2 Classification. Classification of cars is completed via a hump or flat switch movement. Cars in the receiving yard are sorted to a track with cars of the same block or destination. The MIS computer determines the proper classification and records the inventory changes. The process is as follows.

(1) Train humping sequence

- Input
  - Estimated block mix of arriving trains.
  - Estimated time of arrival of arriving trains.
  - Receiving yard inventory and length of stay in receiving yard.
  - Classification track occupancy.
  - Outbound train connections.
- Output--Order of trains to be humped.
- Processing--The simplest method of scheduling the classification of trains is to hump the earliest arriving train first--first-in/first-out (FIFO). A more elaborate scheme is to classify specific trains out of order so as to provide more cars to meet the calling times of outbound trains. If any train has been in the receiving yard over a certain length of time, it would automatically be humped first.

(2) Track assignment

- Input
  - Receiving yard track inventory.
  - Car records of cars to be classified.
  - Classification table--Destination/block number, block number/track.
  - Block or destination inquiry from systemwide MIS or other yard or railroad.
- Output
  - Track assignment for each car in car order.
  - Block or destination inquiry to systemwide MIS or other yard or railroad.
- Processing--If the car record contains a block assignment, a track assignment is made from the classification table. A block assignment may be swung from one track to another if the normal track is full, if the track allocation has been changed (dynamic track assignment), or if the car is a bad order or exceeds a high/wide restriction.

  If a block number has not been assigned and it cannot be made from the destination on the waybill, an inquiry must be made of the receiving railroad (empty cars) or the systemwide MIS computer. Similarly, if the destination of the car is unknown and unavailable from the waybill, an inquiry must be made. Otherwise, the car is classified onto a hold track.

(3) Communication to PC computer

- Output--Track assignment.
- Processing--Transmit to PC computer track assignments for the train to be humped. The file size within the PC computer determines whether track assignments are transmitted immediately before humping or whether a number of assignments can be stored in the PC computer before humping.

3.2.1.3 Inventory Update. Car movements made before departure makeup and made independent of humping or train arrival must be entered into the MIS computer to keep inventory records accurate. The process is as follows.

(1) Inventory update from PC computer

- Input
  - Track location of each car.
  - Misroutes, bad-order cars, hold cars, swing cars.
  - Car weight.
  - Distance-to-couple changes.
  - Track status changes.
- Output
  - Updated car record (location and weight).
  - Updated track record (number of cars, occupancy, status).

- Processing--If only the track location is sent from the PC computer after humping, only exceptions from the switch list need to be returned. Otherwise, a record for each car must be sent, either as a group or as each car crosses the clear point of the classification track and is no longer under the control of the PC computer. This record includes the car weight for each car. If track status and distance to couple are kept by the MIS computer, these records are also passed to it from the PC computer. Changes in weight and car location are made for each car record. The track record is updated with the number of cars, distance to couple, track occupancy, and status changes.

(2) Other inventory update

- Input

  - Car number or old car location
  - New car location.

- Output

  - Updated inventory record
  - Updated car record.

- Processing--When a car is moved in the yard, the change must be recorded in the yard inventory. If terminals are located in the field, the change may be made directly; otherwise, a call is made to the computer operator or to a terminal location to report the car move. This function is also used to reorder cars in inventory if a mistake has been made.

(3) PC alarms

- Input--Alarms from the PC computer.

- Output

  - Notice to operator/users
  - Log file.

- Processing--Alarms and warnings from the PC computer may be sent to the MIS computer for display on terminals or for recording.

(4) Inventory update report

- Input--Inventory changes.

- Output--Log report of changes.

- Processing--All inventory changes either via the PC computer or yard movements should be reported to inform the yard of changes made.

3.2.1.4 Outbound Processing. Once a train departs, inventory records must be changed to reflect the loss of cars. Additional information on the departing train is stored, and information on the train and its contents is forwarded as an advance consist. Car records for departed cars are removed from the active car file and transferred into a history file or logged and then deleted.

An additional function of outbound processing is the monitoring of outbound train preparation including the movement of cars from classification to departure tracks. The outbound processing steps are the following:

(1) Monitor train preparation

- Input

  - Train departure schedule.
  - Tracks/blocks to be pulled for each train.
  - Departure track occupancy.
  - Crew and engine availability.
  - Car record for cars of applicable blocks, including car length and weight.

- Output

  - Crew assignment.
  - Track makeup instructions.
  - Graphic monitoring of departure.
  - Available departure tracks of required length.
  - Engine requirements for weight pulled.

- Processing--A number of techniques can be used to help with departure train makeup. The train departure schedule notifies the trainmaster of train calling times. Classification yard inventory information can then be used to display for the trainmaster the location of cars to be pulled. Departure yard occupancy and the total length of cars to be pulled determine possible departure tracks. The total weight of the train can be used to specify engine requirements. The makeup crew schedule, if known to the MIS computer, can be used to assign crews to the makeup operation and issue crew assignments.

  All this information can be displayed graphically to the trainmaster and/or yardmaster to assist in decision making. As each task or move is completed and the trainmaster is notified, the monitor program is updated to monitor the train preparation. Similar methods may be used to monitor inbound trains.

(2) Prepare consist report and outbound train sheet

- Input

  - Inventory of departing train
  - Car records of departing train
  - Engine and crew of departing train
  - Departure time.

- Output

  - Consist report
  - Outbound train sheet.

- Processing--Individual car records are compiled for the consist report, a printout of the advance consist. The outbound train sheet contains train departure information: tracks pulled, engines used, crew used, cars in train, total length, time called, departure time and track, train symbol, and destination.

(3) Update inventory and departure records

- Input

  - Outbound track
  - Departure time.

- Output

  - Inventory adjusted for departed cars.
  - Departure record in history file.
  - Detailed car records are moved to history.

- Processing--The inventory of the departure track is used to move each detailed car record to a history file after the train departure time is added to the record. The inventory is then updated to reflect the departed cars. A train departure record is created for the train departure.

(4) Advance consist

- Input

  - Consist record
  - Car report.

- Output--Advance consist.

- Processing--The MIS computer or front-end communications processor compiles advance consists for the departing train from the consist report and car records. This information is translated to the format of the outbound consist and of the receiving computer.

3.2.2 Inventory File Structure

The structure of MIS inventory files is determined by available storage space, processor efficiency, and the frequency and type of information inquiries and inventory changes. Most MIS inventory systems are bound by CPU or response time; therefore, the structure of files should allow for efficient compilation of reports and inquiries.

3.2.2.1 Manual Inventory Update. When inventory changes are made within the yard and are not automatically reported (arrival, humping, and departures), they must be manually reported to maintain the accuracy of the yard inventory. Examples include pulls, flat switching, resequencing, coupling tracks, and cars pulled from shop tracks.

If a single car has been moved, its identity and new location must be entered. Car identity can be made by car ID number or by the position of the car in the track (e.g., track 35, line 14). If a car is identified by its ID number, a file in car ID order with a reference to the track location is helpful to find the correct inventory file location to be changed. If the car is identified by track and line number, a file in car ID order is not needed. The new location may also be identified by track and line number (track 35, following car number 16) or by car ID (following car number XXX1234).

If a group of cars is to be moved, the cars may be identified by a range of cars (cars ID1 to car ID7), track location (track 17, cars 15-32), or by relative location (car ID1 plus six cars south or track 17, line number 15 plus 16 cars).

3.2.2.2 Search for Car. Searching for a specific car is difficult if all car records are in inventory order. In that case, the user must search sequentially through all inventory records. If a second file listing cars in car ID order is available, processing time is saved by searching this file.

A method for saving file space is to create a "hashing" algorithm. A well-defined hashing algorithm evenly distributes cars among equal-size "buckets" (file location). When a search is made for a car by ID number, a hashing key is computed and only this bucket need be searched for the car. The algorithm creates artificial classifications that are randomly and evenly distributed, in contrast to actual car ID numbers, which are unevenly distributed because of a number of factors--traffic flow, yard location, time of day, season, and the like.

3.2.2.3 Track Status. An often-requested report is a track report listing cars in inventory order. The frequency of this report and the amount of additional information required influence the amount of information kept in the track-standing inventory file. In addition to car ID, track reports often include location, loaded/empty status, contents, destination, and weight. In this case, if this report is requested often, the track-standing inventory file may be designed to store car ID, loaded/empty status, contents, destination, and weight. Exhibit 3-15 is an example of a track inventory file containing often-required information.

If additional information is also needed for other reports but less frequently, it is usually stored in randomly accessed car record file to save storage space.

3.2.2.4 Levels of File Structure Sophistication. This section discusses the three levels of sophistication used in inventory file structures. The major objective in file design is

EXHIBIT 3-15 SAMPLE INVENTORY FILE

Page 1 Of 1

File Name CLINVFILE Description Cars Located At Potomac Yard & Industrial

Characters per Record 11 Records per Block 45 Revision No.

Date Runs Using This Record 1-INBOUND2

FIELD NUMBER	RELATIVE POSITION	NUMBER OF CHAR-ACTERS	MODE	DESCRIPTION	PROGRAM REFERENCE
1	0	4	X	Car Initial	1
2	4	3	K	Car Number	1
3	7	1	X	L = Load E = Empty	1
				D = Dangerous	1
				X = Explosive	1
				G = Liquified Petro. Gas	1
4	8	1	B	Gross Tons	1
5	9	1	B	Classification Track	1
6	10	1	B	Actual Track	1

Courtesy of Potomac Yard; Richmond, Fredericksburg, and Potomac Railway Company.

to limit file storage requirements while organizing the files in a way that speeds processor searches for frequently used reports and inquiries. Figure 3-2 depicts three file structures.

Minimum Inventory File Structure--With a minimum inventory file structure, limited information reports are available, and only inventory information is accessible. The car record file contains car ID and track location, and it is best stored using a hashing algorithm to permit more efficient searches for the car ID.

The inventory file has a block of storage for each track. The size of each track area is used to record the maximum number of cars on the track; a table gives the address of each track header. This is an inefficient use of storage because partially filled tracks have extra storage space. Inventory information is stored here, as are car ID, block number, length, weight, and loaded/empty/special status.

Car movements are made by track location or by track and car ID. Inventory is a sequential file; therefore, any car deletions or insertions necessitate that the entire file be rewritten. This requires processing and file access time.

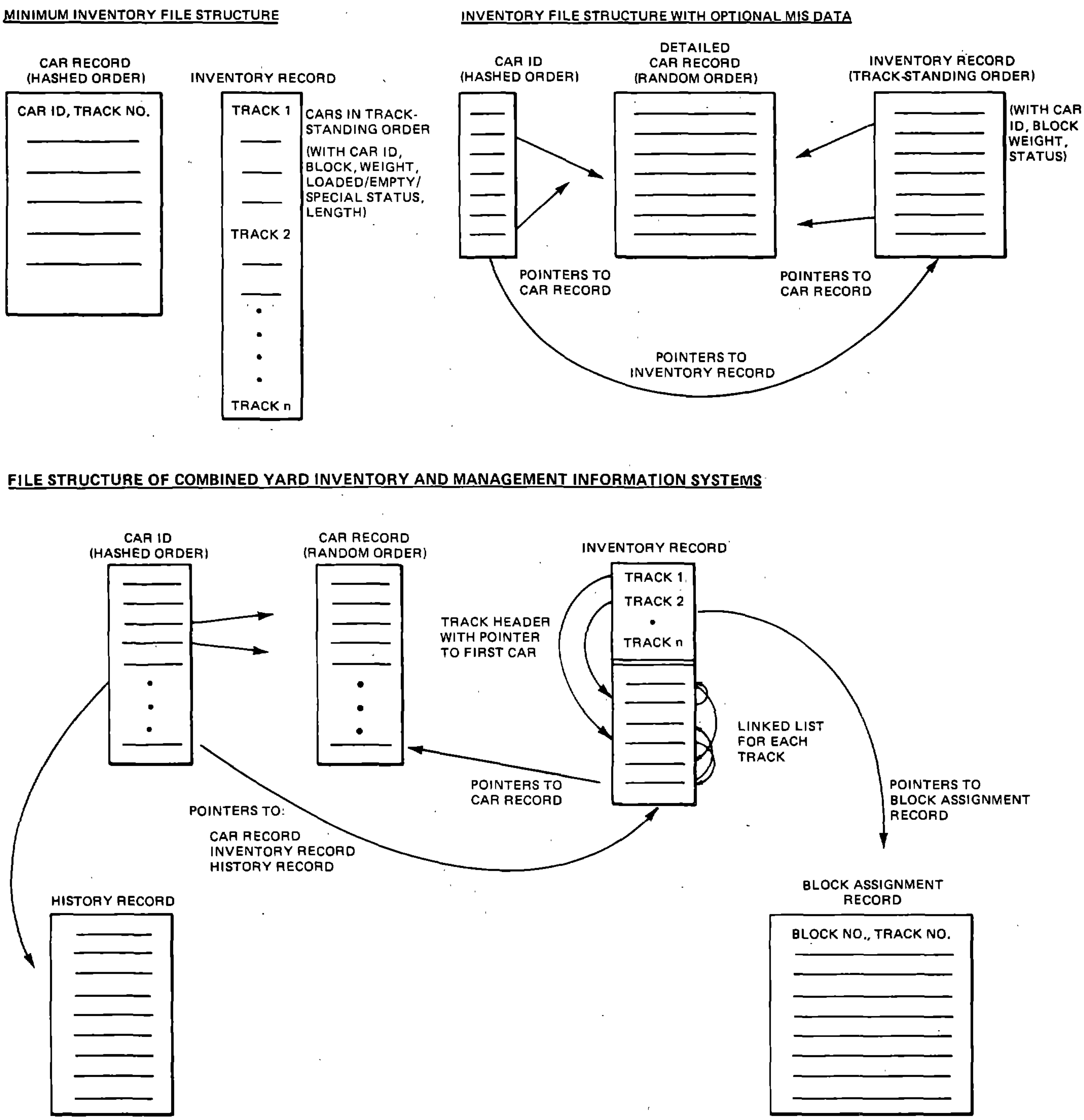


FIGURE 3-2. THREE INVENTORY FILE STRUCTURES

The minimum inventory file structure gives only track location by car ID. All other information is accessed by track location. Inventory listings and switch lists are made efficiently, as they are made in file order and require only one file access. This file structure assumes that few inquiries will be made by car ID. It is most likely to be used with an inventory simulation where events are posted to the yard inventory file only after the physical movement.

Inventory File Structure with Optional MIS Data--An inventory file structure with optional MIS data is designed to act as a yard inventory system, interface with a PC computer, and keep a minimum of MIS information. In a typical configuration with this file structure, the majority of yard MIS data would be stored on the systemwide computer located at a central remote site. Revised switch lists are sent at the end of humping to update the remotely kept inventory.

The car ID file contains car IDs and pointers to car record and inventory files. These records are also hashed into smaller groups for more efficient file searches.

The car record file (Exhibit 3-16) contains most of the required information, including any MIS information. The information is easily accessible from pointers from the car ID and inventory files. Records are randomly stored to save storage space.

The inventory file (Exhibit 3-15) contains car records in track-standing order. This file contains only car ID, weight, block number, and loaded/empty/special status. This is the minimum required to produce switch lists and useful inventory inquiries without accessing the car record file. This system would be expected to have static block-to-track assignments. Because this inventory file structure is an inefficient use of space, a trade-off is made between more efficient data storage and the data stored that are often accessed in inventory inquiries. These accesses would require more processing and file access time if they were located in the car record file. The amount of car data stored in the inventory file is kept to a minimum. Inquiries may be made by either car ID or inventory location, as both have pointers to the car record file. Car movements may be made by track and position, track and car ID, or car ID; the last is the least efficient.

Pointers allow all car information to be more accessible by car ID, inventory, or by sorted subsets by track, weight, and/or block. No yard history is kept. This design uses more complex software, but offers a wider and more flexible range of reports.

File Structure of Combined Yard Inventory and Management Information Systems--This file structure is designed to store more MIS information, provide an easily accessible yard history, and easily change inventory as required by real-time inventory. The car ID file contains ID numbers and pointers to inventory, car record, and history files for each car. Car record files contain car information and more extensive MIS information. The information is accessible from pointers from the car ID and inventory files. A pointer to the block assignment file allows block and track assignments to be changed easily.

The inventory file has a header record for each track containing the track number, block number, length, and length left and a pointer to the first car on the track. Each individual car record contains car location and pointers to the car ID file, car record file, and to the next car on the track. Car movements are made very easily by breaking and remaking pointers in inventory.

The yard history file (Exhibit 3-17) contains in chronological order the car record of those cars that have left the yard within a specific period. The block assignment file contains for each block a description of the block, currently assigned tracks, secondary tracks, and companion blocks. The addition of the block description allows for more frequent, flexible changes as required by dynamic track assignment.

If the inventory is double linked in two directions, inspection lists may be printed easily in either direction. Inventory updates may be made quickly by changing pointers, and this facilitates the upkeep of a real-time inventory--receiving inventory updates as each car passes into the classification yard as reported by the PC computer.

3.2.2.5 Enhancements for File Structure. To enhance file structures, a hashed ID file can be used for fast access, but it uses more memory. Many times, information is only needed for a car in the context of an inventory inquiry. To save file accesses, the information most likely to be required with an inventory inquiry should be stored in the inventory file. If a switch list is to be generated, all the information required should be in the receiving yard inventory file so that no other files need be accessed.

If the car ID is included in the inventory file, a car movement need only be specified by track and ID, not by the specific location on the track. With multiple users accessing and changing the inventory, the actual inventory position on a track may change quickly and the car move may be made on an incorrect car if made by track position only.

Points from the ID file and inventory file may also be used to provide access to car information. This saves storage because the individual car records may be stored randomly. Changes in inventory location therefore do not affect the location of the car record file. This method is a trade-off against keeping some of the information in the inventory file. When the car record file is independent and reached through pointers, it is accessible through either ID or inventory files but is not as easily accessible for inventory inquiries as when the information is in the inventory file. The information most often sought may be kept in the inventory file,

EXHIBIT 3-16 SAMPLE CAR RECORD FILE

File Name MASTERFILE Description CARS LOCATED AT POTOMAC YARD & INDUSTRIALS

Characters per Record 200 Records per Block 4 Revision No. ______

Date ______ Runs Using This Record 1 - INBOUND2 5 - RECOVRIC

FIELD NUMBER	RELATIVE POSITION	NUMBER OF CHARACTERS	MODE	DESCRIPTION	PROGRAM REFERENCE	
	0	1	B	Shop Defect Code	1	5
	1	4	X	Car Initial	1	5
	5	3	K	Car Number	1	5
	8	1	X	L=Load E=Empty	1	5
	9	2	X	Kind of Car	1	5
	11	1	B	Classification Track	1	5
	12	1	U	Railroad In	1	5
	13	1	U	Railroad Out	1	5
	14	1	B	Gross Tons	1	5
	15	1	B	Net Tons	1	5
	16	6	X	Contents	1	5
	22	9	X	Destination City	1	5
	31	2	X	Destination State	1	5
	33	9	X	Consignee	1	5
	42	2	X	Special Instructions	1	5
	44	5	X	Off-Going Junction	1	5
	49	4	X	Off-Going Road	1	5
	53	3	K	Origin Station Number	1	5
	56	1	X	Shop Reported Code	1	5
	57	1	X	X=INPCFILE Prepaid/Collect	1	5
	58	1	K	Waybill Month	1	5
	59	1	K	Waybill Day	1	5
	60	2	K	Billing Road Code	1	5
	62	3	K	Waybill Number	1	5
	65	9	X	Shipper	1	5
	74	9	X	Origin City	1	5
	83	2	X	Origin State	1	5
	85	4	X	Final Road	1	5
	89	1	X	Routing Code - Agent Shipper	1	5
	90	1	U	Move Type	1	5
	91	1	X	Keypunch/CRT Operator	1	5
	92	6	X	Arrival Symbol	1	5
	98	1	X	Arrival Motor Initial	1	5
	99	2	K	Arrival Motor Number	1	5
	101	1	K	Arrival Sheet Number	1	5
	102	1	K	Arrival Hour	1	5
	103	1	K	Arrival Minute	1	5
	104	1	K	Arrival Month	1	5
	105	1	K	Arrival Day	1	5
	106	1	K	Interchange Hour	1	5
	107	1	K	Interchange Minute	1	5
	108	1	K	Interchange Month	1	5

Courtesy of Potomac Yard; Richmond, Fredericksburg, and Potomac Railway Company.

EXHIBIT 3-16 SAMPLE CARE RECORD FILE (Concluded)

File Name MASTERFILE Description CARS LOCATED AT POTOMAC YARD & INDUSTRIALS

Characters per Record 200 Records per Block 4 Revision No.

Date Runs Using This Record 1 - INBOUND2 5 - RECOVRIC

FIELD NUMBER	RELATIVE POSITION	NUMBER OF CHAR-ACTERS	MODE	DESCRIPTION	PROGRAM REFERENCE
	109	1	K	Interchange Day	1 5
	110	1	X	D Code	1 5
	111	1	X	U Code	1 5
	112	1	X	I Code	1 5
	113	4	X	Piggyback Initial	1 5
	117	3	K	Piggyback Number	1 5
	120	4	K	Standard Trans.Commodity No.	1 5
	124	1	B	Actual Track (Location)	1 5
	125	5	X	Departure Symbol	1 5
	130	2	K	Departure Motor Number	1 5
	132	1	K	Ordered Hour	1 5
	133	1	K	Ordered Minute	1 5
	134	1	K	Ordered Month	1 5
	135	1	K	Ordered Day	1 5
	136	1	K	Departure Sheet Number	1 5
	137	40	X	Routing From SCL	1 5
	177	4	K	Placed Year,Month,Day,Hour	1 5
	181	4	K	Pulled Year,Month,Day,Hour	1 5
	185	2	K	Industrial Patron/Locator	
				Code	1 5
	195	5	K	Shifted Year,Month,Day,Hour	
				Minute	1 5

EXHIBIT 3-17 SAMPLE YARD HISTORY FILE

Page 1 Of 2

File Name CARRECFILE Description Departed Cars

Characters per Record 200 Records per Block 4 Revision No. ______

Date ______ Runs Using This Record 5-RECOVRIC

FIELD NUMBER	RELATIVE POSITION	NUMBER OF CHAR-ACTERS	MODE	DESCRIPTION	PROGRAM REFERENCE
	0	1	X	Shop Defect Code	5
	1	4	X	Car Initial	5
	5	3	K	Car Number	5
	8	1	X	L = Load E = Empty	5
	9	2	X	Kind of Car	5
	11	1	B	Classification Track	5
	12	1	U	Railroad In	5
	13	1	U	Railroad Out	5
	14	1	B	Gross Tons	5
	15	1	B	Net Tons	5
	16	6	X	Contents	5
	22	9	X	Destination City	5
	31	2	X	Destination State	5
	33	9	X	Consignee	5
	42	2	X	Special Instructions	5
	44	5	X	Off-Going Junction	5
	49	4	X	Off-Going Road	5
	53	3	K	Origin Station Number	5
	56	1	X	Shop Reported Code	5
	57	1	X	X=INPCFILE Prepaid/Collect	5
	58	1	K	Waybill Month	5
	59	1	K	Waybill Day	5
	60	2	K	Billing Road Code	5
	62	3	K	Waybill Number	5
	65	9	X	Shipper	5
	74	9	X	Origin City	5
	83	2	X	Origin State	5
	85	4	X	Final Road	5
	89	1	X	Routing Code - Agent - Shipper	5
	90	1	U	Move Type	5
	91	1	X	Keypunch/CRT Operator	5
	92	6	X	Arrival Symbol	5
	98	1	X	Arrival Motor Initial	5
	99	2	K	Arrival Motor Number	5
	101	1	K	Arrival Sheet Number	5
	102	1	K	Arrival Hour	5

Courtesy of Potomac Yard; Richmond, Fredericksburg, and Potomac Railway Company.

EXHIBIT 3-17 SAMPLE YARD HISTORY FILE (Concluded)

File Name CARRECFILE Description Cars Located at Potomac Yard & Industrial

Characters per Record 200 Records per Block 4 Revision No. ______

Date ______ Runs Using This Record 5-RECOVRIC

FIELD NUMBER	RELATIVE POSITION	NUMBER OF CHAR-ACTERS	MODE	DESCRIPTION	PROGRAM REFERENCE
	103	1	K	Arrival Minute	5
	104	1	K	Arrival Month	5
	105	1	K	Arrival Day	5
	106	1	K	Interchange Hour	5
	107	1	K	Interchange Minute	5
	108	1	K	Interchange Month	5
	109	1	K	Interchange Day	5
	110	1	X	D Code	5
	111	1	X	U Code	5
	112	1	X	I Code	5
	113	4	X	Piggyback Initial	5
	117	3	K	Piggyback Number	5
	120	4	K	Standard Trans. Commodity No.	5
	124	1	B	Actual Track (Location)	5
	125	5	X	Departure Symbol	5
	130	2	K	Departure Motor Number	5
	132	1	K	Ordered Hour	5
	133	1	K	Ordered Minute	5
	134	1	K	Ordered Month	5
	135	1	K	Ordered Day	5
	136	1	K	Departure Sheet Number	5
	137	40	X	Routing from SCL	5
	177	4	K	Placed: Year,Month,Day, Hr.	5
	181	4	K	Pulled: Year,Month,Day, Hr.	5
	185	2	K	Industrial Patron/Locator Code	5
	195	5	K	Shifted: Yr,Mo,Da,Hr,Min	5

while MIS and secondary information might be kept in the car record file.

Dynamic allocation of inventory file space may be used to save the excessive amount of storage required by having a static memory record for each possible physical inventory relocation. The first method is to have a minimum space for car records in inventory order, for example, space for 40 cars for each track. When more than 40 cars are in a single track inventory, a second inventory module must be added by providing a pointer in the last record of the first module.

A second method uses a linked list for each individual car. As each car is added to inventory, the last car's pointer points to the memory location of the new car. When a car is deleted or moved, the pointer of the car preceding is changed to point to the car following it. This method no longer requires that the inventory file or module be rewritten upon every insertion or deletion.

The most common applications of the PC computer in hump yards are automatic routing and switching of cars and automatic speed control. The PC computer may perform many other functions in the yard, however, including routing trains from the receiving yard to the hump, generating the pin-puller's list, detecting bad-order cars, controlling hump engine speed, keeping the track inventory and car location records, issuing operational reports, aligning switches and providing area protection during trim operations, and determining power requirements for makeup operations. Exhibit 4-1 lists common PC functions.

4.1 RAILROAD OPERATIONAL DESCRIPTION

4.1.1 Receiving Yard to Crest

Trains in the receiving yard are ready for humping once the cars have been inspected and the brake systems have been bled. Train lists are updated in the MIS computer or in both the MIS and PC computers to reflect any changes in the makeup of the trains. These lists should be accurate before the switch list and pin-puller's list are compiled.

4.1.1.1 Generating a Switch List. A central (as opposed to a yard) computer may assign a classification code to each car in the train based on the classification table (which contains the classification tracks and the various criteria for assigning a car to a particular track) and waybill information (destination, consignee, deliver-to-road, and so forth) (Martin & Durand, 1974). The yard MIS computer translates this code into a classification track assignment, which is passed to the PC computer in the form of a switch list. Alternatively, the classification table may be transmitted to the PC computer from a central data computer, from which it generates classification track assignments for cars.

4.1.1.2 Generating Pin-Puller's List. A pin-puller's list is developed from the switch list and track assignment instructions. The number of cars in a cut is determined by the destination of contiguous cars and the maximum allowable cut length. The pin-puller's list may be in hard-copy form with written annotations or it may be a CRT display, which allows changes to be made until cars have passed the pin-puller.

4.1.1.3 Routing from Receiving to Crest. The hump controller enters humping requests (i.e., which receiving tracks to pull) at a terminal. Cars affected may be flagged to indicate they are in transit to the crest. The PC computer can align the receiving yard switches to route a train to the hump if the MIS computer has given it the receiving track inventory, current switch settings, and the order in which trains are to be humped. The hump conductor may visually verify that the switch list and actual train makeup agree.

4.1.1.4 Bad-Order Car Detection. The PC computer can also be programmed to stop the humping process if a bad-order car is detected. A dragging equipment indicator installed on the hump lead sends a warning signal to the computer when dragging equipment deflects the vertical steel plates of the indicator, closing an electrical contact (Petracek et al., 1976). Inspectors can enter car numbers into the PC system via terminals located at inspection points along the hump lead; bad-order cars are then automatically switched to repair tracks. Instruments to detect broken flanges, loose wheels, and overheated bearings may also be installed and interfaced with the computer.

4.1.1.5 Hump Engine Speed Control. Radio remote control can be used to regulate the speed of the hump engine. A requested speed is transmitted via radio to the locomotive's on-board control system. The requested speed is compared with the actual speed measured by axle tachometers to control propulsion, braking, and emergency systems (DeIvernois, 1972). The requested speed may be constant, or it may be varied to maximize yard throughput by not having the speed set for the worst-case situation (i.e., where a trailing car is likely to catch up to preceding car if cars are humped too fast) (Petracek et al., 1976). Another method of controlling locomotive speed is to transmit recommended speed via radio signals to the engine, where it is displayed to the engineman in digital form or as a colored light (e.g., red = stop, green = fast, yellow = slow, flashing red = back up).

4.1.2 Crest to Classification Tracks

4.1.2.1 Automatic Routing and Switching. The switch list usually is transmitted from the MIS computer to the PC computer. The PC computer should have a switch list storage capacity equal to the standing capacity of the receiving yard. If there is no provision for a stored switch list, an operator may enter the track destinations at a terminal when humping occurs or read them off punched cards. As a car goes over the crest, its ID number is removed from the CRT display in the control tower.

Knowing the classification track destinations, the computer progressively aligns the switches for each cut as the cut travels through the switching area. As a cut rolls forward, track circuit repeaters or loop circuit repeaters associated with each switch detect the cut's presence and the switch is thrown in accordance with the assigned track destination. Wheel

EXHIBIT 4-1 COMMON PROCESS CONTROL FUNCTIONS

A. Receiving Yard to Crest

- Routing to crest
- Switch alignment
- Automatic B/O (bad order) detection (hot box, dragging equipment
- Automatic height detection

B. Crest

- Pin-puller's list or display
- Hump engine speed--radio control
- Weight for inventory record and speed control

C. Automatic Routing and Switching

- Classification guide received from MIS or manual input
- Cut length detection
- Car identification and verification
- Automatic switching--precrest to clear point
- Automatic swing to B/O or rehump track or when track closed
- Car location update
- Alarms, correction/avoidance
- Distance-to-couple, occupancy measurement

D. Automatic Speed Control

- Weight scale input
- Weight classification
- Distance-to-couple input
- Weather input
- Calculate rollability
- Master speed calculation for single and multiple car cuts
- Master retarder control (feedback)
- Exit group speed calculation for single- and multiple-car cuts
- Group retarder control (feedback)

E. Manual Routing and Switch Alignment

- Manual reroute switching
- Add or delete missing or extra cars in inventory (inventory adjustment)
- Swing car to B/O or rehump
- Swing when track closed
- Set switches for backing over the hump
- Manual hump engine control

F. Manual Override for Speed Control, Route Selection, and Switching

G. Trim End

- Switch alignment
- Control of inert skate
- Track lock/unlock (last switch set away from blue-flag track)

H. Returned to MIS Computer

- Weight or weight class for each car
- Distance to couple for each track
- Inventory update (flag misroutes and B/O, hold, and swung cars)
- Final switch list update or individual update as car clears
- Track status (lock/unlock)

I. Alarms

- Misroutes
- Track overflow
- Speed control errors
- Equipment failures
- Catch-up and cornering conflicts
- Stalled cars
- Short track circuit
- Missing, extra, out-of-sequence cars
- Editing changes to resequence cars
- Track swing required (overflow or locked track)
- Hazardous car to be humped
- Car out of order

J. Reports

- Hump use
- Total number of cars humped
- Updated final classification list (note bad order and hold/height exceed cars)
- Log of manual operations
- Log of stalls, misroutes, track swings, cornering/catch-up conflicts, and extra, missing, and resequenced cars
- Speed distribution report and log of speed control errors
- Log of equipment failures
- Log of track overflows
- Swing assignments
- Height restrictions for each classification

K. Inquiries

- Track status (locked, blue flag)
- Track occupancy (distance to couple)
- Associated swing assignment (B/O and rehump tracks) for each classification track
- Hazardous cars

detectors or car presence detectors are used to prevent switches from operating under long cars or cuts that span the track circuits. A wheel detector sets up a magnetic field in a section of rail. When a wheel passes, it changes the field, inducing a current in a nearby coil. The electronic detection of this current indicates that a wheel has passed. The computer counts the pulses from the wheel detectors and compares them with the known number of axles (which were counted by wheel detectors near the crest). A presence detector senses that a car is over a switch by detecting a shift in frequency caused by the car passing over a coil that controls the frequency of an oscillator (Petracek et al., 1976).

In addition to ensuring that it does not throw a switch under a car, the computer checks for cornering possibilities and stalled cars (cars stopped short of the clearance point) before setting switches. If either of these situations exists, the cars following are routed to a fouling track and a misroute is reported. In the case of cornering,* the following car may be routed to the same track as the preceding car, and a misroute is reported.

High cars, wide cars, and overweight cars can be automatically routed to special tracks unless the hump conductor manually overrides that assignment and sends the car to the track assignment of its original destination. The hump conductor's console may have various pushbuttons for routing cars to loaded repair, empty repair, hold, and clean-out tracks. Any misroutes or swings are logged and then reported to the MIS computer.

4.1.2.2 <u>Automatic Speed Control</u>. To determine the correct exit speed of a car from a retarder, the computer combines the following information: car acceleration, car weight, distance to couple, track grade and curvature, route information, and weather and wind conditions. Track grade and curvature and route information are already in the computer at the time of humping. Field sensors are used to collect the other measurements.

When a cut of cars rolls down the hump, wheel detectors or photocells sense its presence and signal the computer to monitor the radar output (Williamson et al., 1973). Radar measures car speed by sending out a beam of radio waves and measuring the reflection of those waves off the moving car using the principle of Doppler frequency shift. Velocity measurements are taken ahead of the master and group retarders. An electronic coupled-in-motion scale measures the weight of the cars. Each axle load is measured separately and then summed to obtain the gross weight of the car.

*Leading and trailing corners of two consecutive cars collide because of insufficient separation at a branching point.

Distance to couple is determined by the use of track circuits. The axle acting as a shunt across the track circuit, the impedance of a classification track is measured from the clearance point to the nearest axle of the car that last entered the track. The impedance of the rails varies directly with the distance from the circuit origin to the nearest shunt, so correlating impedance with distance to couple is possible (Petracek et al., 1976). For cuts longer than a specified length, an arbitrary rolling resistance factor is applied, based on the number of axles and weight. The length above which a rolling resistance cannot be obtained depends on the length of constant grade between the crest and master retarder and the intermediate and group retarders. Photosensors at the crest determine the length of a cut.

4.1.2.3 <u>Car Location</u>. PC systems track cars as they travel from the crest to their respective destinations in the classification yard using bidirectional wheel sensors. The PC inventory may be stored until all cars in a train are humped, at which time it is sent to the MIS computer in the form of an "as-humped" list. Alternatively, inventory update records may be transferred to the MIS computer as each car reaches its designated classification track. Sensor inaccuracy, mistakes in pin-pulling, and the like cause discrepancies between sensor-obtained tracking information and the switch list and pin-puller's list. Corrections are entered to reflect the true location of the cars. In some cases, the PC computer provides such MIS functions as an up-to-date listing of all cars in each of the receiving and classification tracks and an up-to-date record of total tonnage and number of cars on each receiving and classification track. A record of each car may also be kept in track entry order and switch list format.

4.1.3 <u>Classification Track to Departure Track</u>

4.1.3.1 <u>Align Pullout Route</u>. If the trim-end supervisor wishes to pull a cut of cars between the classification and the departure tracks, he can enter this request on the PC computer from an on-line terminal. Signals are sent to align the switches between the classification and departure tracks. Providing there is no conflict with a protected or blocked area, the switches are aligned. The supervisor releases the area lock after the pulling operation so that other pulling operations can occur.

4.1.3.2 <u>Makeup Requests</u>. Via an on-line terminal, yardmasters enter train makeup requests specifying classification track numbers from which trains are to be made up, initial and number of the first and last cars of the block, location of the transfer in the yard, and available road engine power. The computer then updates affected inventories and prints updated inventory summaries in the yard office. Gross car weights (determined during the humping operation) are added (starting with the first car of the first classification track specified) until the total weight equals available engine power. If the yardmaster has requested too many or too few cars to be pulled by available power, the PC computer reports this. The PC computer also reports if the requested power for the total weight is exactly correct. The yardmaster can then obtain additional power, add cars, or proceed as planned, depending on the report he receives.

Bidirectional wheel sensors can be used to count axles as cars are pulled from the classification yard. The computer notifies the pullout clerk if the car tracking data do not agree with data already in the computer. The clerk can then take the necessary corrective actions after viewing the move by television camera or on videotape (Martin & Durand, 1974). This function can also be performed by the MIS computer (Section 3.1.7).

4.1.4 Operational Reports

The PC computer can be designed to produce a number of operational reports, including:

- Misroutes.
- Full track.
- Speed errors.
- Equipment failures.
- Destination change.
- Total time of hump use.
- Number of cars humped.
- Final switch list.
- Updated switch list after humping.
- Swing assignments.
- Classification track assignments.
- Operation logging.
- Pullout query.
- Technician and yardmaster reports.
- Height restrictions for each classification.
- Updated group tally.
- Receiving yard tally.
- Classification track inventory.
- Inventory summary for yardmaster.
- Departure yard log.

4.2 COMPUTER SOFTWARE FUNCTIONAL DESCRIPTION

4.2.1 Operating System: Executive Program

The operating system hardware controls the internal operation of the computer hardware and software systems. The operating system is also called the executive program, system monitor, or system supervisor. It is a collection of programs, many of which are provided by the computer vendor and refined or augmented by the user. The real-time operating system for the railroad classification yard PC application is a specialized and often yard-dependent system. It should perform the following duties:

- Accept interrupts from I/O devices, time clock operator, and so on and maintain job request queues.
- Schedule and allocate program execution times in accordance with the priority levels of the programs and the resources available.
- Allocate spaces in both main memory and secondary memories and perform file management (i.e., program loading, overlay, linkage, and so forth).
- Perform utility functions, such as operator interface, failure detection and management, accounting for computer utilization, file protection, and the like.

Figure 4-1, a conceptual representation of a yard PC computer software system, indicates the relationship between the real-time operating system and the other software subsystems.

The software functions of a PC computer in a hump yard may be considered from two standpoints: (1) the geographical area of the yard or (2) the nature of the function. Geographically, the PC computer covers the following areas:

- Precrest (from receiving yard to the crest).
- Crest.
- Crest to master retarder.
- Master retarder to group retarder.
- Group retarder to tangent retarder.
- Tangent retarder to coupling point.
- Pullout end and departure yard.

The PC computer software can be grouped according to:

- Sampling routines
- Data reduction routines

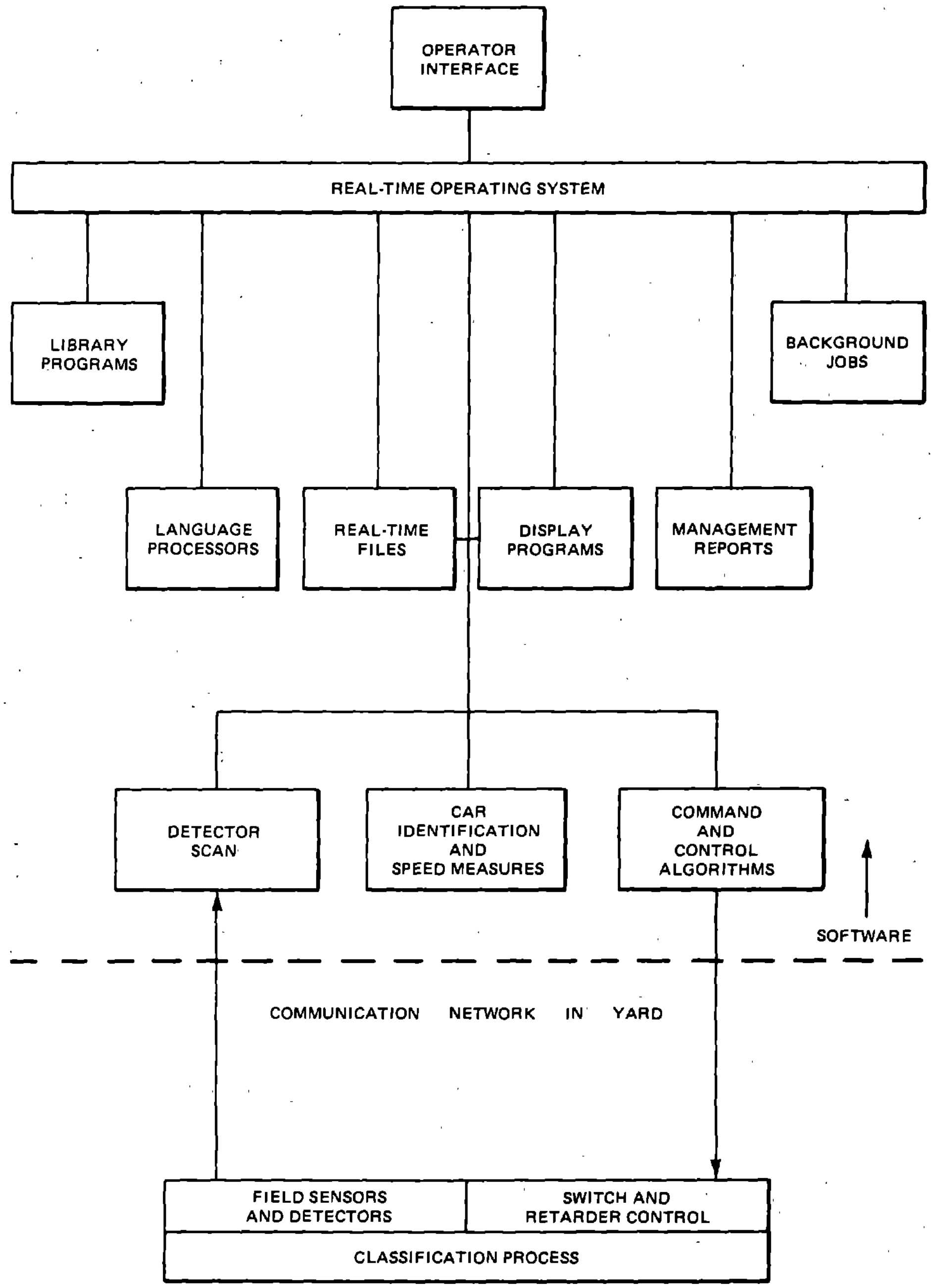


FIGURE 4-1. CLASSIFICATION YARD PROCESS CONTROL SOFTWARE SYSTEM

- Control function routines
- System monitoring and MIS routines.

Depending on the field devices under the PC computer's surveillance and control, the sampling technique can either be on a fixed-time increment basis (cycling) or on an event-occurrence basis (interrupt). The determination of which technique to use is a trade-off between data resolution and computer resource cost.

The data reduction software is essentially a continuation of the sampling routines, whereby raw input data are translated into information useful for decision making. For example, the activation of a wheel detector is translated, along with other input information (e.g., system clock time), into car speed.

The control function routines generate control signals from both input data (reduced) and prestored control parameters and decision rules.

The system monitoring routines manipulate and organize the existing input and output information into statistical and management reports.

The software packages currently in use are briefly described below.

4.2.2 Precrest Control

The precrest control function covers the processing of train arrival information that is relevant to the automatic alignment of the switches from receiving tracks to the hump approach. In this function, the information on arriving trains and certain operational management decisions (e.g., the classification table, the switch list) are transferred from the MIS computer to the PC computer. ACI reading and/or receiving inspection reports will correct any last-minute discrepancies that have existed between the train consist and the physical inventory (Section 3.2.1.1).

While the inventory data of the receiving area are being verified and corrected, the operational management decision function of the MIS is generating track assignment decisions based on preestablished criteria, classification tables, decision rules, and the current and predicted track utilization of the yard. The track assignment determination is a necessary step for generating the switch list, pin-puller's list, and the routing sequence from the receiving tracks to the hump. At this time, the PC function receives the switch list from the MIS function. The PC computer performs automatic routing from the receiving tracks to the hump and maintains precrest car inventory.

This function is usually supported by two major routines: the routine for automatic routing from the receiving tracks to the hump crest and the receiving yard inventory routine. Table 4-1 summarizes the input, process, and output for these routines.

4.2.3 Crest Control

The crest control function extends from the receiving leads through the crest. The principal function is to ascertain humping sequence, effect accurate pin-pulling, measure and control hump speed, control and verify proper routing, and initiate corrective actions when necessary. Under this function are eight software routines (see Table 4-1):

- Route and engine control.
- Determining optimal hump speed.
- Generation of pin-puller's list.
- Hump wheel detector input processing.
- Hump speed calculation.
- Hump operating display.
- Switch alignment for front-end trim operation.
- Area trim protection.

4.2.4 Crest to Master Retarder

The purpose of the postcrest PC function is to check and verify pin-pullers' actions, collect car velocity and other physical data for speed control, and update the car inventory. This function is aided by wheel detectors, photocells, radars, weight scales, and so forth.

The typical routines (see Table 4-1) are as follows:

- Postcrest wheel detector processing.
- Photocell detection routine.
- Radar detection of cut length.
- Determination of car axle count.
- Posthump car identification and verification.
- Weight scale routine.
- Weight classification.

4.2.5 Master Retarder to Group Retarder

(1) General

The PC computer functions from master retarder to group retarder are to check and verify pin-pullers' actions, collect car velocity and other physical data for speed control, and update the car inventory in the posthump file. As cuts enter and leave the master retarder track section, the speed control function detects and measures the presence of a cut and its velocity and determines retarder control action. At the same time, the software function monitors changes in track inventory. Although the routing of cars and switch alignment are determined at the hump, an elaborate set of switch logic software routines is required in the posthumping process to prevent misroutings, collisions, and derailments.

(2) Master retarder control

- Input
  - Axle passage detection from wheel detectors placed before the master retarder.
  - Velocity reading from entrance radar unit.
  - Car speed control information (e.g., weight class, cut length, height).
  - Requested release velocity.

Table 4-1

SOFTWARE ROUTINES FOR PROCESS CONTROL COMPUTER

Routines	Input	Process	Output
PRECREST CONTROL			
Routine for automatic routing from receiving yard to crest	Receiving track inventory Switch list Track assignments list Current switch settings	This routine is to align the switches so as to move the next batch of cars from the receiving tracks to the hump in accordance with humping sequence.	Command to align receiving yard switches for the next batch of cars heading for the hump
Receiving yard inventory routine	Train consists ACI data Receiving inspection report	This function can either be performed by the MIS computer or the PC computer, or both. If automatic routing from the receiving track to the hump crest is to be done by the PC computer, a requisite is that the PC computer have on-line access to the inventory file by car, by track, and by position.	Car ID by track and by position to be maintained in the PC computer on-line file
CREST CONTROL			
Route and engine control	Switch alignment settings from receiving tracks to crest Hump speed instruction from dynamic speed control calculation or from a predetermined average speed Current hump signal status from the hump approach track circuit Automatic routing plan as generated by precrest control	This software routine is to ascertain whether the route switches from receiving tracks to hump approach are aligned properly, in accordance with the automatic routing plan generated by precrest control route selection software. The control signal is sent out in one or more of the following ways: • Go or no-go signal to the hump shove light. This is the simplest form of control command. • Simple speed control instructions to the shove lights, e.g., stop (red), fast (green), or slow (amber). • In a more automated system, the speed con-control instruction goes directly into the cab engine control.	Go or no-go signal to hump shove lights Speed category number to shove lights Speed control command to cab
Determining optimal hump speed	Desired yard throughput parameters Penalty factors (as a function of the routing and attributes of the cut) in case of misrouting due to catch-ups	This routine helps to increase the throughput of the yard by assigning the highest hump speed feasible; yet at the same time it must minimize the probability of misroutes due to to car catch-ups. The cost of correcting misroutes often negates the intended throughput gain. The determination of hump speed depends on the route pattern selected (distance between switches, track layout, etc.), the tag number of the cut (e.g., if the current cut goes to the same classification track as the previous cut, the two cuts may be relatively close to each other), and many other speed control and rollability considerations to be determined with inputs from other parts of the yard.	Speed control instructions to hump engine

Table 4-1 (continued)

Routines	Input	Process	Output
Generation of pin-puller's list	Track assignment list Switch list Post-humping surveillance feedback	From the switch list and track assignment instructions (an operational management decision under the MIS function), a cut list is developed. A cut is determined by the common destination of contiguous cars and the upper limit on the number of cars allowed in a cut. The pin-puller's list is traditionally in hard-copy form, with written annotations and modifications. In a more dynamic system the pin-puller's list is a CRT display, and the system can accept last-minute changes or corrections as long as the cars have not passed the pin-puller. The cars that are not to be humped (but will be rerouted) are also displayed.	Pin-puller's list (static) or CRT display (dynamic)
Hump wheel detector input processing	Signals representing passage of axle over each sensor Time of each passage	The sensing of axle passage over the wheel detectors generates a priority interrupt to the PC computer. The computation of axle count involves the time of passage of each axle and the elapsed times between axle passage times, as well as the correlation of passage data between wheel detectors. Usually two sets of detectors are placed at the crest to enhance computation (particularly in a bidirectional movement situation) and provide redundancy checks.	Net count for each wheel detector Net car count by direction
Hump speed calculation	Data from hump wheel detector processing routine	This routine is to compute hump speed based on the passage times of the first and second axles and the assumed length between axles. Conceptually, the hump speed (in feet per second) is the difference between the two axle passage times (in seconds) divided into the wheelbase (in feet).	Current hump speed
Hump operation display	Data from hump speed wheel detector processing routine Switch list	The hump wheel detector process routine gives confirmation that a car has passed the crest. This information is used in conjunction with the switch list to establish the identification of the car that has just been humped. As each car passes the wheel detectors, its ID is removed from the top of the CRT display, and the whole CRT page moves up one line. The purpose of the display is to give visual verification of the accuracy of the on-line record. A mistake may occur if the wheel detection routine miscounted the axles or if the switch list is out of sequence relative to the physical order of the cars. The cars that passed the visual verification go into the "as-humped" file.	Updated "as-humped" list CRT display

Table 4-1 (continued)

Routines	Input	Process	Output
Switch alignment for front-end trim operation	Instruction from hump supervisor's console Computer flags indicating blocked or locked areas	On occasion the hump engine performs trim operations at the front end of the yard (from the crest to classification tracks), such as for rehumping. The instruction for front-end trim comes from the hump supervisor, and the switch alignment is done automatically for an area, provided the trim operation will not conflict with another protected or blocked area at the time. When the trim operation is completed, the hump supervisor releases the area lock, and the switches go back to automatic routing control.	Control signals to align switches for an area between the crest and a track
Area trim protection	Instructions from yardmaster or area supervisor via on-line console	Requests may come from the yardmaster or area supervisor via on-line computer terminals for performing trim operations in an area (defined by tracks, leads, frogs, etc.). This routine identifies all the switches leading into the defined area and locks them out from external control until a release command is issued. During the area lock, the area is not accessible to automatic routing or humping. The control is given to the area supervisor or foreman for local manual or semiautomatic control. The significance of the area lock to the humping operation is that the locked tracks may necessitate that some cars be rerouted to slough tracks.	Switch alignment instructions for locking all switches leading into the trim area for area protection. The switches within the trim areas are released from automatic control and turned over to area supervisors for local control.
POSTCREST CONTROL			
Postcrest wheel detector processing	Signals from wheel detectors (placed between the crest and the master retarder) indicating the passage of an axle. The input comes into the computer on a priority interrupt mode.	This is a front-end processing routine to reduce wheel passages and passage times to axle counts and length between axles [based on a measured (or assumed) speed]. To enhance detection reliability, two sets of wheel detectors usually are installed (between the crest and the master retarder). This information will be compared with photocell and radar data for validity check.	Cumulative axle counts for each detector Passage time of each axle
Photocell detection routine	Photocell beam contact changes create an interrupt to the PC computer Switch list	The photocell detects the beginning and the end of a cut (the coupling joints constitute a continued state). The passage times of the front and end of a cut will help to verify wheel detector data and enhance identification of car inventory between the crest and the master retarder. The photocell signal also is used to turn on and off the radar Doppler pulse accumulator for car length measurement. Another function of the photocell sensor is to detect the height of cars for rollability measurement.	The determination of the beginning and the end of a cut and the sending of a signal to the radar unit to turn on and off the radar Doppler pulse accumulator for car length measurement Cut passage times (both front and end) Car height category

Table 4-1 (concluded)

Routines	Input	Process	Output
Radar detection of cut length	Radar pulse return	The radar Doppler pulse accumulator is turned on and off by the photocell's detection routine. Cut length data are calculated and used for speed control and car identification.	Cut length measurement
Determination of car axle count	Wheel detector routine information Photocell routine information Radar routine information	On a priority interrupt basis, wheel detector data (axle passage and passage time), photocell data (beginning and end of a cut, coupled cars), and radar data (cut length) are checked and correlated to determine the number of axles for each car. An assumption is made about the standard distances between axles, distance from coupling point to the first set of axles, the maximum number of axles on a car, and so forth.	Determination of the number of axles for each car and length between axles Car speed
Posthump car identification and verification	Output from the determination of car axle count routine Switch list Pin-puller's list	This routine reconciles car and cut information in the switch list and pin-puller's list with the sensor-obtained information by confirming the number of cars in each cut, car lengths, cut lengths, and so forth. In the event of a discrepancy (may be caused by sensor inaccuracy or mistakes in pin-pulling), corrective entries are made to reflect the true tag number of cars. A crest-to-master retarder car inventory file ("as-humped list") is maintained in order of car position and cut configurations. Each car record also contains data on speed control and rollability functions.	Updated car and cut identification in crest-to-master retarder inventory file
Weight scale routine	Car axle count and identification information Weight of car Car record	The scale (placed after the car ID and speed sensors) obtains the gross weight of each car and, based on other attributes from the car record (car type, capacity, tare weight, length, height), the routine determines whether the car is overweight. If so, the car is rerouted. The gross weight of the car is recorded in the car record for two purposes: for waybill information under the MIS function and for speed control.	Enter scale weight to car record in the "as-humped" list for revenue and speed control purposes Initiate rerouting in the event the car is overweight
Weight classification	Weigh rail reading	The weigh rail's function is to classify each car into a weight class (e.g., light, medium, heavy) to activate master retarder control. This function is a locally distributed PC function that may be handled by either an analog device or a microprocessor. Its output will go directly to a master retarder control. It can be viewed as an inner loop of a speed control function (the outer loop being the scale weight routine processed by the PC computer for a more global application).	Weight category classification to master retarder control

- Axle passage detection from wheel detectors placed after the master retarder.

- Velocity reading from exit radar unit.

- Output

  - Entrance speed.

  - Retarder control command.

  - Speed control information to be passed on to the next control point.

  - Car ID and movement information to be transferred to the next control point.

  - Failure messages and initiation of failure-mode operation (a result of detection or radar failure).

- Processing--The PC computer for retarder control (whether it is for master, group, or tangent) has either a centralized or a distributed design. Under a centralized design, the mainframe PC computer accepts raw input data from field sensors and then computes and sends retarder control commands directly to each retarder control. The computation is based on a global speed control objective function and the integration of sensor input information from all the zones and areas under the control of the total PC system. Under a distributed design, each retarder is controlled by a separate PC unit (e.g., an analog controller or a microprocessor) that is often installed locally at the retarder site. The local PC unit functions as the "inner loop" of an overall PC system. Figure 4-2 is a schematic of a typical distributed retarder control system.

  The wheel detectors and radar units provide input to the retarder control system. Besides velocity measures, the field sensors also provide axle passage time and count, which are necessary for correlating car passage information with car ID and associated speed control descriptions (weight class, height, length, etc.) as measured during previous PC functions.

  The speed control logic then determines the desired release velocity of the cut and calculates the control setting of the retarder. The actual exit velocity information is then used for speed control functions elsewhere in the yard.

  The master retarder control routine also checks the reliability of the wheel detectors. If the detector is determined to be malfunctioning the routine generates failure messages and initiates failure-mode actions for a possibly impeded PC operation.

(3) <u>Car location update routine (or car following routine)</u>

- Input--Car passage information (both entrance and exit) from wheel detectors placed before and after the master retarder.

- Output--Updated car location record (i.e., change car location from in front of the master retarder to in back of the master retarder).

- Processing--Although this is a specific function to update car position records from the postcrest zone to the post-master retarder zone, the software routine should be a generalized yardwide car-following system that moves a car record from one zone to another as long as the zone boundaries are defined and the car movement information is transmitted to the PC computer. In many yards, the PC computer maintains individual car records only from the crest to the clear point. Car records for the rest of the yard (receiving area, pullout end, departure area, etc.) are maintained by the MIS computer.

(4) <u>Switch logic for conflict avoidance</u>

- Input

  - Wheel detector input to indicate the entrance or exit of a cut.

  - Track and switch layout parameters of track circuits that require special attention.

  - Car location information on those cars being moved from the crest to the classification tracks.

  - Actual switch position as read by roadside equipment.

- Output

  - Control signals to switches including the application of fixed time delays (if necessary).

  - Routing error or conflict warning messages to flag a potential problem condition. The routing error should trigger corrective actions in the automatic routing and car location software.

  - Indicator for switch obstruction condition when the actual state of the switch differs from the requested position.

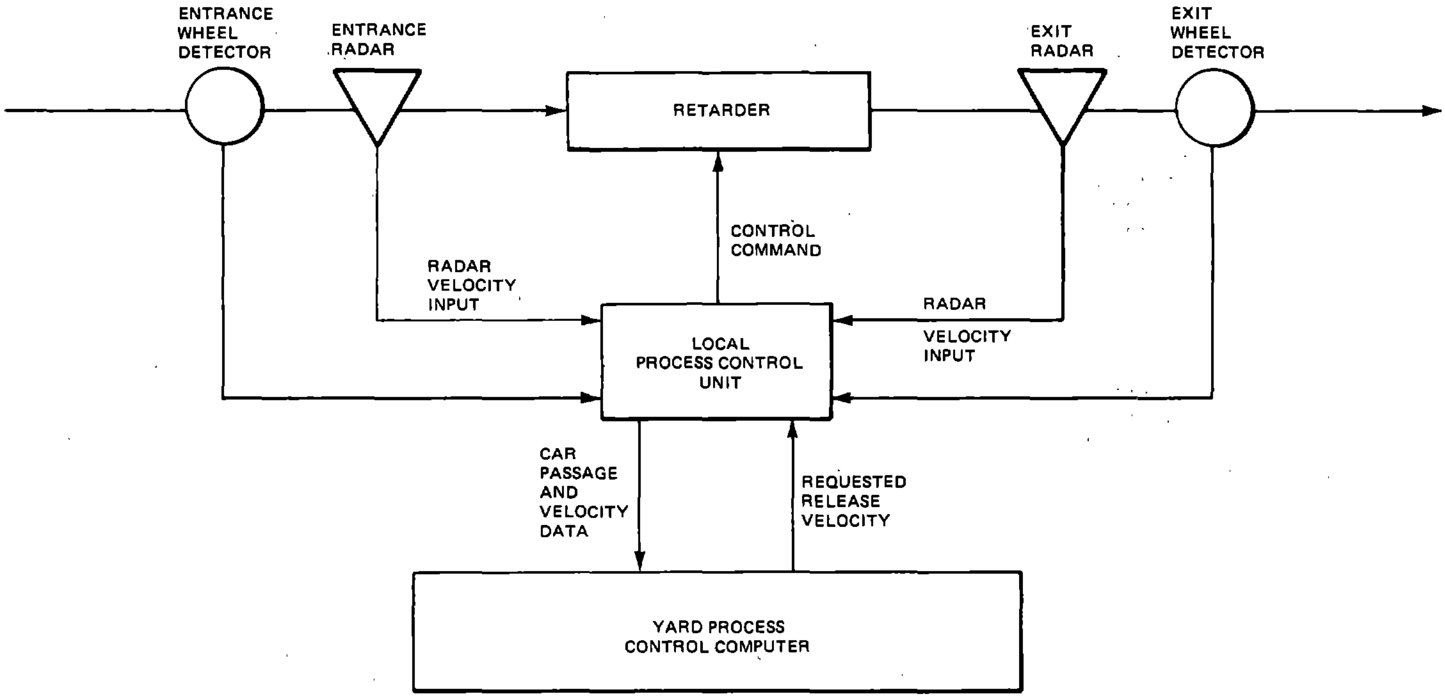


FIGURE 4-2. EXAMPLE OF A DISTRIBUTED RETARDER CONTROL SYSTEM

- Processing--This switch logic is to forestall any possible car catch-ups or conflicts due to limitations of the track/switch configuration, exceptional physical characteristics of cars, or abnormal speed dynamics that may result in collisons, cornering conflicts, or derailment. Although described in this section, these logical routines are applicable at almost any area from the master retarder to the coupling point. The potential conflict situations and the methods of solving them differ from yard to yard. The following are some examples of these situations:

  - Car Catch-up--A catch-up condition develops when a successor cut enters a switch track circuit before the predecessor cut clears the circuit. If the second cut does not share the same switch setting as the first, it ends in the wrong track circuit and a routing error therefore exists. Corrective entries are generated for the subsequent routing plan of the misrouted cut. Also, the car-location record must reflect this error.

  - Cornering Conflict--If the distance between two consecutive cuts at a branching point is very close and the velocity of the second cut is sufficiently great relative to the first cut, a potential cornering situation exists. If a collision is preferred over cornering, the software routine generates instructions to leave the switch unchanged for a safety margin time delay after the passage of the first cut at the switch point.

  - Car Stall--The stalling of a cut after a switch may cause a cornering accident if the switch is thrown and a second car passes. In this situation, a logic may be chosen to leave the switch open to the track circuit of the first cut and divert subsequent traffic from arriving by properly adjusting upstream switch settings.

  - Short Track Circuit--When a car is much longer than a switch track circuit (e.g., the track circuit is 60 feet long and a car is 100 feet long), the switch must not be thrown until the car completely clears the track circuit.

  - Multiswitch Zone--Certain zones are defined by more than a single switch. This is very prevalent when two or more switches are very close and they must be thrown in tandem to safely route a cut. In this case, the switch logic generates a special control command for the zone (or zones) affected.

- Check Switch--The switch software also verifies the actual switch position after a switch change command has been given. If the state is not changed as requested, adjustment entries must be made until the obstruction is removed.

4.2.6 Group Retarder to Tangent Retarder

The software routines governing the area from the group retarder to the tangent retarder are very similar to those in the other areas. The differences are mostly in the physical characteristics of the switch/track layout, speed control parameters, and the input fidelity of field sensors. The software routine for group retarder control can be viewed as identical to that for master retarder control--the type of wheel counting devices, velocity measurement devices, and retarder PC functions are the same. The car-following routine (to move the car record from zone to zone) and switch logic routines also are the same for this area.

4.2.7 Tangent Retarder to Coupling Point

The retarder control, car counting and location updating, and switch control of the tangent retarder to coupling point area are similar to those for the other areas. The principal differences are in the switch/track layout geometry and the characteristics of field sensors (e.g., the number of wheel detectors or the number of radar units). The only major additional control function in this PC area is the detection of car movement and car stop location. The car movement detection routine is briefly described as follows:

- Input
  - Move or no-move input from track space equipment.
  - Car ID and previous speed control data.
- Output
  - Position of car stoppage
  - Coupling velocity of each car
  - Track occupancy estimate.
- Processing--After a cut passes through the tangent retarder section, the PC system must know where it stops on the track. This can usually be detected by using track space equipment for motion or no-motion indications. The changing consecutive space readings imply the motion of a cut from one section of the track to the next. A constant state among two or more consecutive space readings indicates a stoppage of the cut. The speed with which these readings are changed forms the basis for estimating coupling velocity, which is used for the calibration of speed control parameters upstream. The stop position of the cars is used to update car inventory and car location records. The stop position can also be estimated using a distance-to-couple track circuit (Section 4.1.2.2).

4.2.8 Pull-out End

(1) Exit (skate) retarder control

- Input
  - Input from lock-unlock routine.
  - Detection input from track circuit to indicate the exit of last car in a pullout operation.
- Output--"Open" command to exit retarder. The retarder is normally closed.
- Processing--The skate retarder is usually closed. It opens during the pullout operation (when the track is locked from the crest end) and closes after the last car clears the track circuit.

(2) Track lock-unlock routine

- Input
  - Request from hump supervisor
  - Exit retarder open indicator
  - Alarm condition indicator
  - Car movement data.
- Output
  - Track lock instructions for switches
  - Track unlock instructions for switches.
- Processing--Except under alarm conditions in the yard, most of the locking and unlocking instructions are results of humping or pullout operations. The processing is mainly generated by a request from the area supervisors or the hump supervisor.

4.3 LEVEL OF AUTOMATION

Although fully automated process control is the de facto standard for new and recently upgraded classification yards, PC systems vary greatly from yard to yard. The major determining factors in these variations are economic considerations, present and anticipated throughput, and the age* of the yard. Significant variations

---

*In new yards with high traffic volume, a fully automated system may have been installed. In older yards that have been upgraded, often a combination of manual, semiautomatic, and automatic operations is found.

based on the level of automation include: semiautomatic switching/manual retarding and semiautomatic switching/automatic retarding.

The semiautomatic switching system is known as a relay system. Pushing a track destination button (or buttons) sends a destination code to switching circuits, which align switches for the cut as it proceeds to its assigned classification track. A certain number (usually four or five) of track destinations can be stored at one time in the storage unit in the hump conductor's office; additional relay storage units are in front of lap and individual switches. Switch repeater relays route the destination code of each cut to the next appropriate storage unit (the one in front of the next switch to be thrown) as the cut shunts the track circuit preceding each switch along its route. As each cut's destination code is transferred from the first relay storage unit (in the hump conductor's office) to the next storage unit, the hump conductor may enter the destination of another cut (GRS, 1957).

With manual retarder control, a retarder operator, located in a building alongside the retarders, manually controls the retarder force and the duration of the force. He uses his experience, observes the car's rolling behavior, and considers the type and often the weight of cars in determining his control action.

With semiautomatic switching/automatic retarding, one computer is used to route cars and produce fixed exit speeds from retarders. The engine foreman, rather than the computer, has the switch list. When his train is ready to be classified, the engine foreman goes to a control console in the hump tower and punches in the classification track assignment for each cut as it approaches the crest using destination codes obtained from the switch list.

In this type of system, the computer systems automatically control the retarders.

This chapter describes analysis methods for systems design in railroad classification yards. The goals of systems design are to analyze yard requirements, design alternative solutions, select feasible options, and define functional and hardware specifications. The implementation following system design is discussed in Chapter 6.

Because process control (PC) systems and management information systems (MISs) have different purposes and end goals, their designs also differ. The PC application has traditionally been considered reasonably straightforward, and because of the problem of interfacing with yard field hardware (e.g., scales, switches, retarders, track circuits), its design has most often been left to the signal manufacturers or systems vendor with little design effort by the yard. (The most notable exception is Santa Fe's designing of the PC equipment at Barstow Yard.) In analyzing process control, railroad designers gain a concrete idea of their own requirements before approaching a system manufacturer. Having a detailed functional specification, a preliminary design, and an estimate of hardware requirements, the railroad can better specify the system required from a turnkey vendor. Thus, the vendor/supplier need not do preliminary design work. In the end, both railroad and supplier benefit from a better system design that meets the actual needs of the yard.

MISs in most cases are designed and installed by the railroad itself, with perhaps the assistance of consultants and vendors. The design and specification of the system must therefore be completed in much greater detail than is required for a PC system. Functional specifications must be further refined into a detailed software design specification. This document is eventually used in program design and coding. The hardware study results in a detailed hardware specification document, which is used as part of the request for proposals (RFP) sent to hardware vendors and is used for the design of acceptance specification and testing. Test plans and implementation schedules must also be completed in detail.

The design of the yard computer system is divided into four steps: the survey phase, analysis phase, software design phase, and the hardware study. Figure 5-1 indicates the relationship between these steps. The design cycle need not proceed step by step; tasks may overlap or be completed concurrently. The concept of a structured systems design is one of the many popular systems analysis techniques used. Others include the BIAIT, BSP, PSL/PSA, and SADT methods (Townsend, 1980). Structured design stresses the use of graphic tools to represent functions of the computer system (Page-Jones, 1980; Yourdan, 1978; Auerbach, 1979). In each phase of the design, the computer system is partitioned

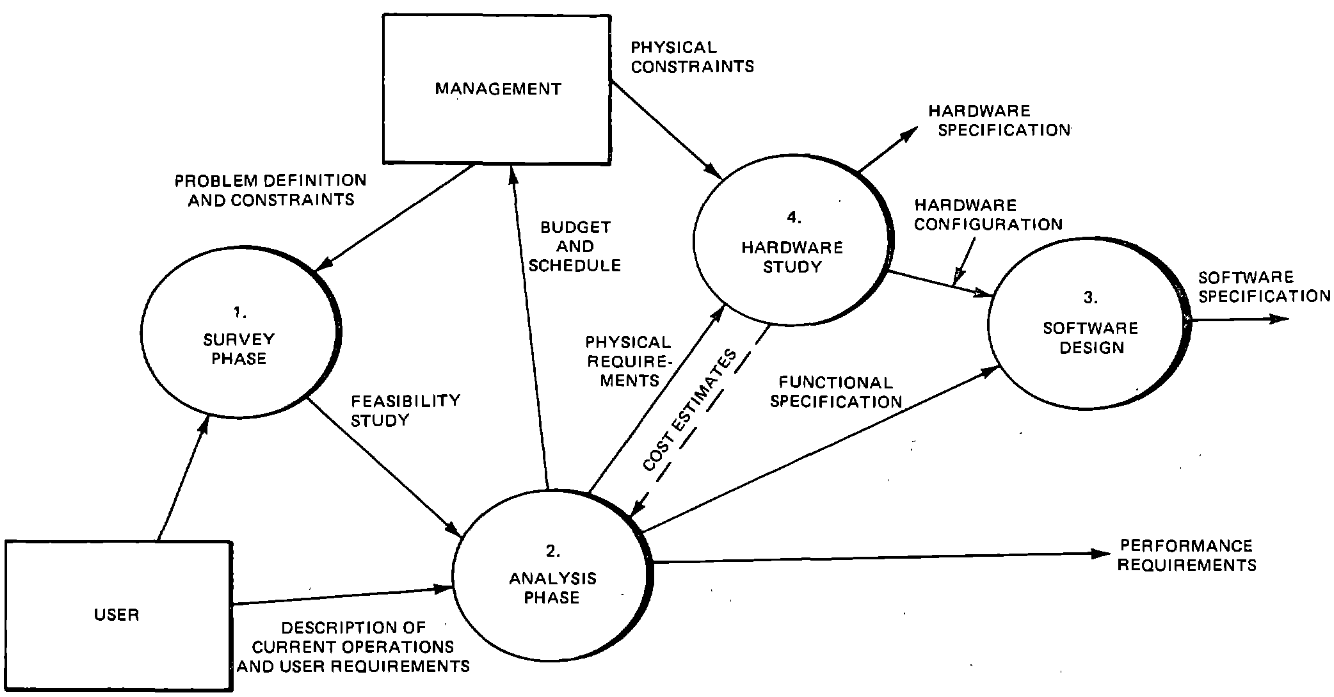


FIGURE 5-1. SYSTEM DESIGN CYCLE

and data flow diagrams are used to represent, in a simplified form, input, output, and function.

5.1 SURVEY PHASE

In the survey phase or feasibility study, yard engineers explore alternative procedures and equipment to decide whether the likely benefits justify the expense of the changes to be made. Constraints from management will structure the scope, budget, and schedule of the project. A preliminary staffing plan (Section 5.1.1) and preliminary schedule (Section 5.1.2) are also established. These are revised throughout the analysis phase. Discussions with knowledgeable users provide a description of the environment and its problems. The feasibility study provides a guide for systems design and includes: (Hartman, 1968)

- Management recommendations, goals, and a description of project charges.
- A general description of the organization.
- A summary of requirements, objectives, constraints, and current problems.
- Overall project schedule including project team staffing and project delivery date goals.

5.1.1 Organization Planning and Staffing

The organization of the systems project team depends on a number of factors, such as: the size and complexity of the proposed system, whether it is to be completely new or a modification of an existing system; whether the computer system configuration is to be decentralized, regionalized, or centralized; and whether the development responsibility is centralized or within the yard. Nonetheless, the first step in planning and staffing any systems project team is to select the project manager. The manager, in turn, selects personnel from both the EDP department and other departments or divisions.

Project leaders are usually chosen from within the EDP department for MISs, but they may be chosen from the signal department for PC systems. Project teams include people with expertise in systems design, systems software, data base management and design, and operations. User representatives for the team are chosen from other departments and from each yard if a centralized or regionalized system is planned. Representatives should be at least at the supervisory level and have a thorough knowledge of the operations that will be incorporated into the new system. In PC system, for example, the yardmaster is likely to be included on the team to provide information on and ideas for system design and specifications. User representatives not only become members of the project team, but also assign operating personnel under their supervision to work with members of the project team, as required.

The entire project team changes in size and composition during the project life cycle. From the development of software and hardware specifications to software development and testing, the number of EDP personnel grows and then usually tapers off as the acceptance test and systems installation point is reached. Concurrently, the user involvement increases.

5.1.2 Development and Implementation Planning and Scheduling

After the project team has been organized, it must specify the tasks to be performed during development and implementation and prepare a checklist of tasks. The task list depends on the type of system to be implemented, but the basic activities are the following:

- Complete feasibility study.
- Complete systems design.
- Develop hardware and software specifications.
- Develop acceptance test plan and tests.
- Complete procurement process, including any benchmark testing required for vendor selection.
- Develop and test software.
- Prepare system documentation.
- Train users.
- Prepare site preparation plan.
- Convert to new system.
- Run acceptance test and install system.

The task list should include not only the prescribed activities, but also a realistic evaluation of the time required to perform each activity (both in terms of calendar days and person-days of work) and the name or title of the person who will be responsibile for the activity.

The implementation and installation cycles for MISs and PC systems in most cases are different because PC systems are less likely to be fully developed in-house. Usually, only larger railroads have the in-house expertise required to develop specialized hardware and software. Therefore, PC systems are usually purchased as a package from an independent vendor.

Because most railroads have some degree of MIS capability, most yard MIS computer systems will be an upgrade of an existing system or they must link to a centralized systemwide MIS computer. The development cycle of an MIS upgrade is the same as that for a new system but must take into account interface and conversion details in the hardware study, acceptance test, and implementation stages.

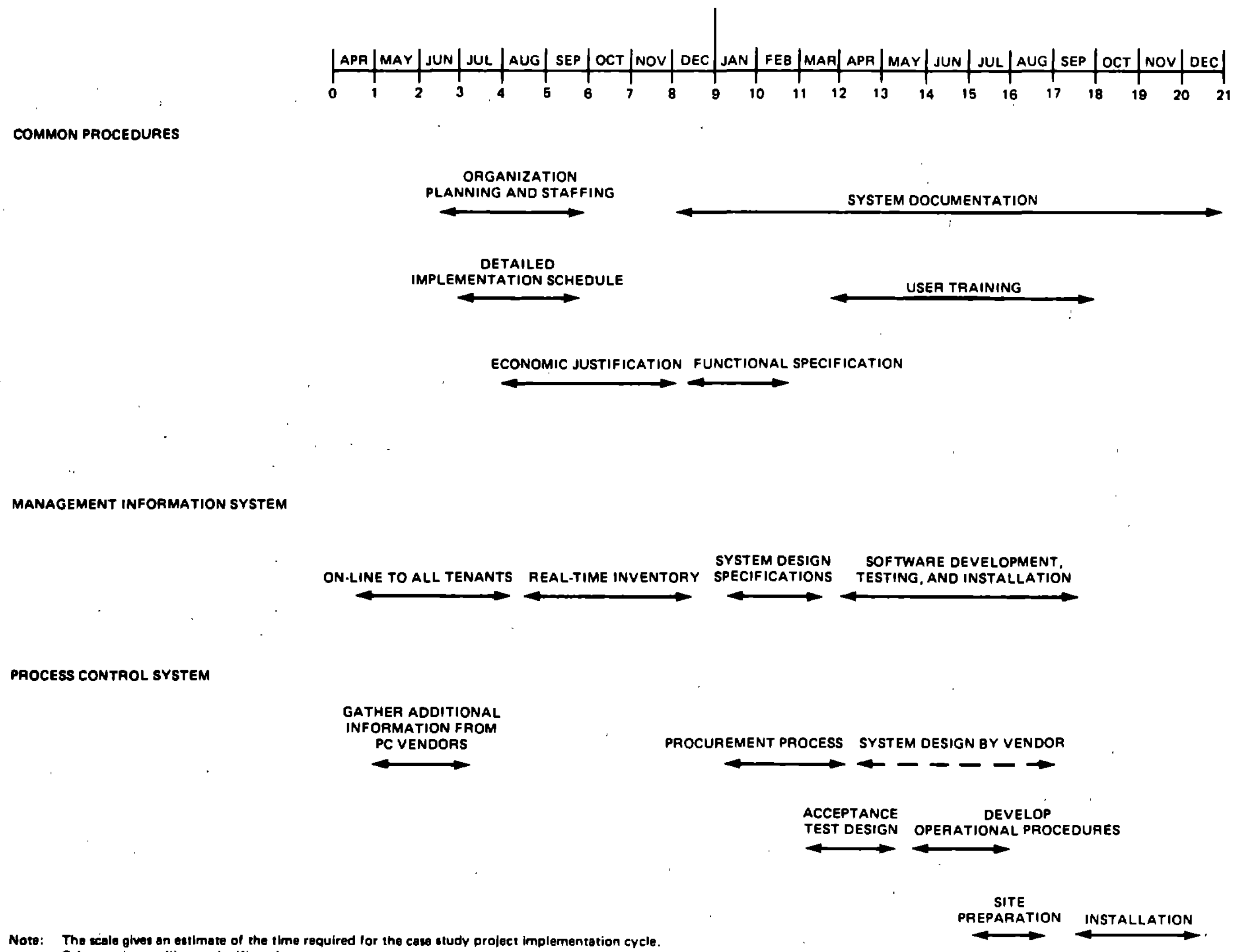
PC system development requires organization planning and staffing, implementation scheduling, and the development of functional specifications. In many cases, detailed design need not be carried further because final design is often a task of the vendor. The remainder of the project involves procurement, design monitoring, user training, and site preparation.

The case study at Potomac Yard involved the procurement of PC hardware and software and the development of additional MIS software, but it required no additional MIS hardware. Therefore, as Figure 5-2 indicates, software changes were scheduled to begin immediately. If additional MIS hardware had been required, it would have had to be installed before software coding began. This is because existing systems are often too small for development or incompatible with the replacement hardware and operating system.

After the task list has been completed, the project team prepares a preliminary PERT diagram, critical path diagram, or other project control charts showing elapsed times and start and completion dates for all the specified activities. This ensures that all necessary activities are performed in the proper sequence and on time. After the critical path and total project time are known, iterations of the chart will probably be made to adjust the time frame by changing schedules or manpower assignments. Figure 5-3 is an example of the critical path diagram from the case study.

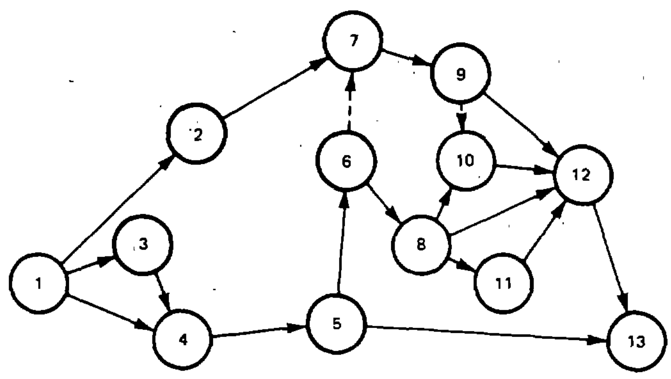
5.2 ANALYSIS PHASE

The analysis phase further defines the project within the goals specified previously in the feasibility document or project charter. Users provide additional information on the present



Note: The scale gives an estimate of the time required for the case study project implementation cycle. Other projects will vary significantly.

FIGURE 5-2. IMPLEMENTATION CYCLE



EVENT NUMBERS	ACTIVITY
1 – 2	Direct communications to each tenant railroad
1 – 3	Organization planning and staffing
1 – 4	Gather additional information from PC vendors
2 – 7	MIS real-time inventory
3 – 4	Detailed implementation schedule
4 – 5	Economic justification study
5 – 6	Functional specifications
5 – 13	System documentation
6 – 8	PC procurement process
7 – 9	MIS system design specifications
8 – 10	Develop PC operational procedures
8 – 11	PC system design by vendor
8 – 12	PC acceptance test design
9 – 12	MIS software development, testing, and installation
10 – 12	MIS and PC user training
11 – 12	PC site preparation
12 – 13	PC installation

FIGURE 5-3. SAMPLE CRITICAL PATH DIAGRAM

system, and the project team explores the feasibility of alternative methods to implement changes and decides which ones to pursue. A description of the recommended system is produced as a functional specification.

In the analysis phase, the project team develops a preliminary hardware configuration design. This is used for cost/benefit estimates but, more important, it becomes the basis for the hardware study. Additional configurations and performance details are developed in the software design phase.

Structured analysis is used for analyzing user requirements, selecting a system that best provides solutions, and documenting a structured description of the system. Figure 5-4 illustrates the seven steps of structured analysis: (1) model current system, (2) derive logical equivalent, (3) model new logical system, (4) develop alternative functional configurations, (5) measure costs and benefits, (6) select best option, and (7) package results as a structured functional specification. The results of the analysis phase include the structured functional specification and guidelines for hardware requirements. Other results may be: a tentative equipment configuration, a performance document, a tentative design, an RFP (depending on detail required), file and data layouts, resource analysis (disk and main memory), and more detailed project scheduling and personnel planning. This information is needed for budget and scheduling estimates for management.

The goal of the analysis phase is to document in a readable and understandable form the functions and goals of the new system. The analysis provides both a link between user needs and the design of changes to meet these needs. In addition, the analysis provides predictions of benefits, scheduling, and performance characteristics. The functional specification is used to communicate the system design to both users and management. The goal is to produce a design that is compliant with both users' needs and management's constraints. The functional specification should be sufficiently comprehensive to use during implementation to evaluate the successful completion of system goals. To remain valuable in later phases of the software design cycle and in the implementation and installation cycle, the functional specification and documents that evolve from it must be easy to modify as required from user feedback and changes in overall goals and constraints. In general, because functional specifications depict only the logical flow of the system, they should be unaffected by changes in hardware, vendors, and operational personnel.

5.2.1 Assessment of Current System

The first step in the analysis phase of systems design is to learn about the current system and its environment. The analysis of current procedures is used to create a model of current functions without regard to expected changes in the yard environment and functions performed. The volume of current and expected yard traffic influences eventual hardware and software design. Information on the current hardware directly relates to the hardware study. Railroad and yard objectives, management structure, and institutional considerations are not quantifiable but influence the design and selection of system design options. Data on current costs are used in the cost/benefit analysis.

The structured analysis method relies on data flow diagrams to model the current physical data flow. For example, current clerical operations can be diagramed on the basis of the movement of waybills, as shown in Figure 5-5. Additional data on the physical environment are also gathered as part of the current system model. Data such as current traffic distribution, yard layout, and hump speed are required for the PC specification.

The operation of all areas of the yard that may affect the eventual computer system must be examined and documented. Areas examined include clerical procedures, present computer and communication systems, communication within the yard, communication outside the yard, inventory control and record systems, management reporting, yard traffic volumes, classification procedures, current automatic car control in the yard, current yard layout, and the current track hardware.

5.2.1.1 Information Gathering. The gathering of information on a classification yard may be organized in terms of system functions. The description and volume of the functions of yard

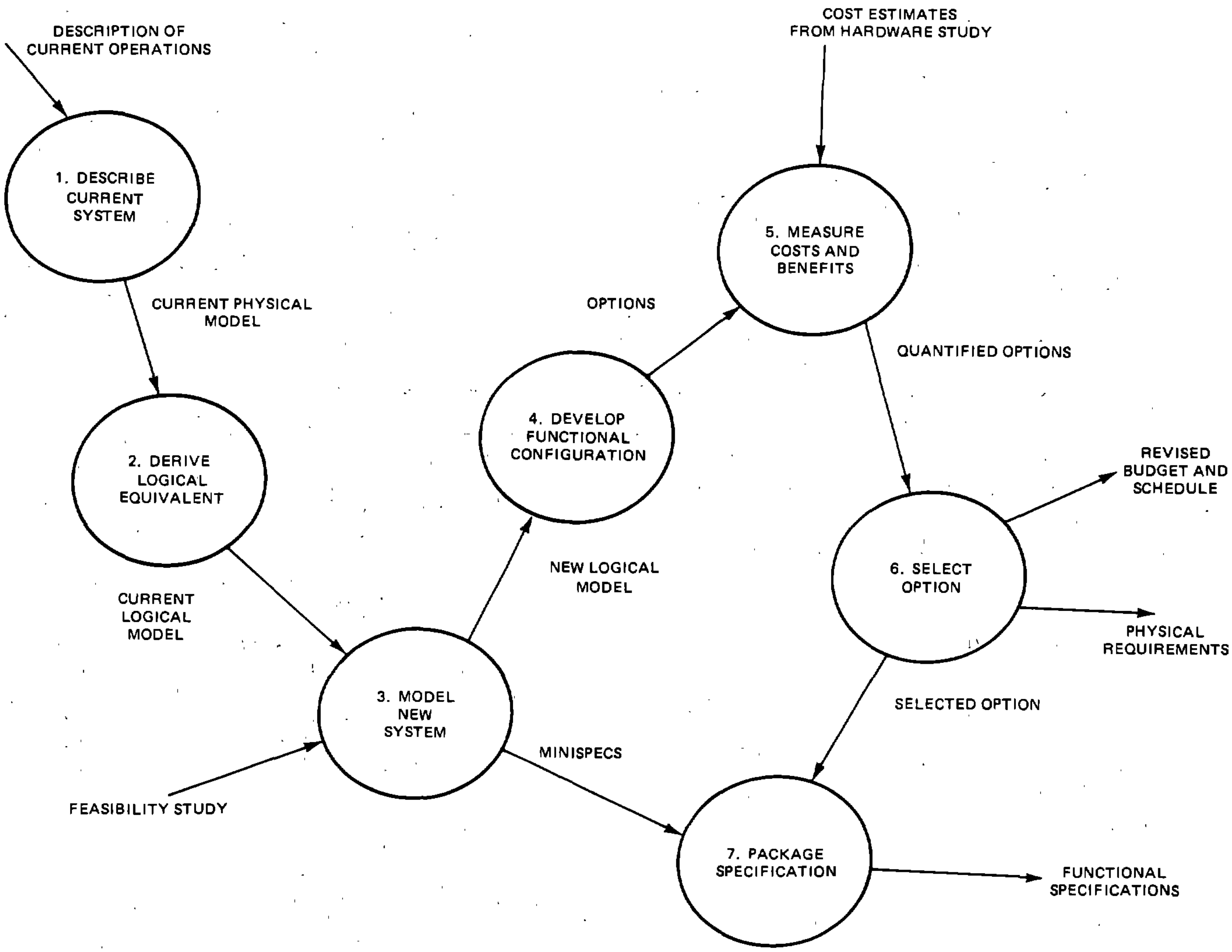


FIGURE 5-4. STRUCTURED ANALYSIS STEPS

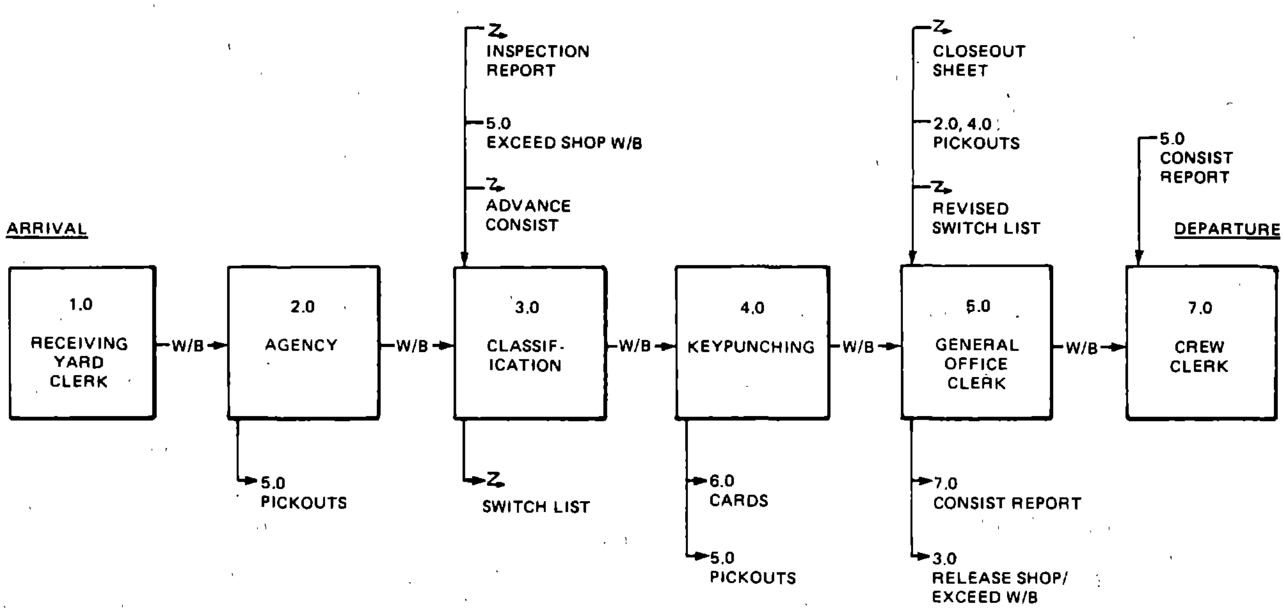


FIGURE 5-5. FLOW DIAGRAM OF WAYBILL PROCEDURES FROM CASE STUDY YARD

inventory and PC systems are used to design models of both the current and future computer systems. The functional model of the yard evolves into the functional specifications used in software design. The volume descriptions are used to determine file requirements and the size of hardware components in the hardware study.

Yard volume quantifiers may define: (1) a physical attribute, such as the number of tracks and their lengths; (2) a car processing volume, such as distribution of arrivals; or (3) the size of a specific computer function, such as reporting frequency. Each function is described by a number of the numerical quantifiers. Each quantifier, in turn, may affect a number of functions. For example, the average number of cars humped as a train is a quantifier for the hump control function, and it also affects the switch list generation function, the PC-to-MIS communications function, and the yard inventory update function. A matrix of functions and quantifiers, such as that represented in Figure 5-6, provides, in terms of discrete data, a complete description of the yard, including interrelationships between each function.

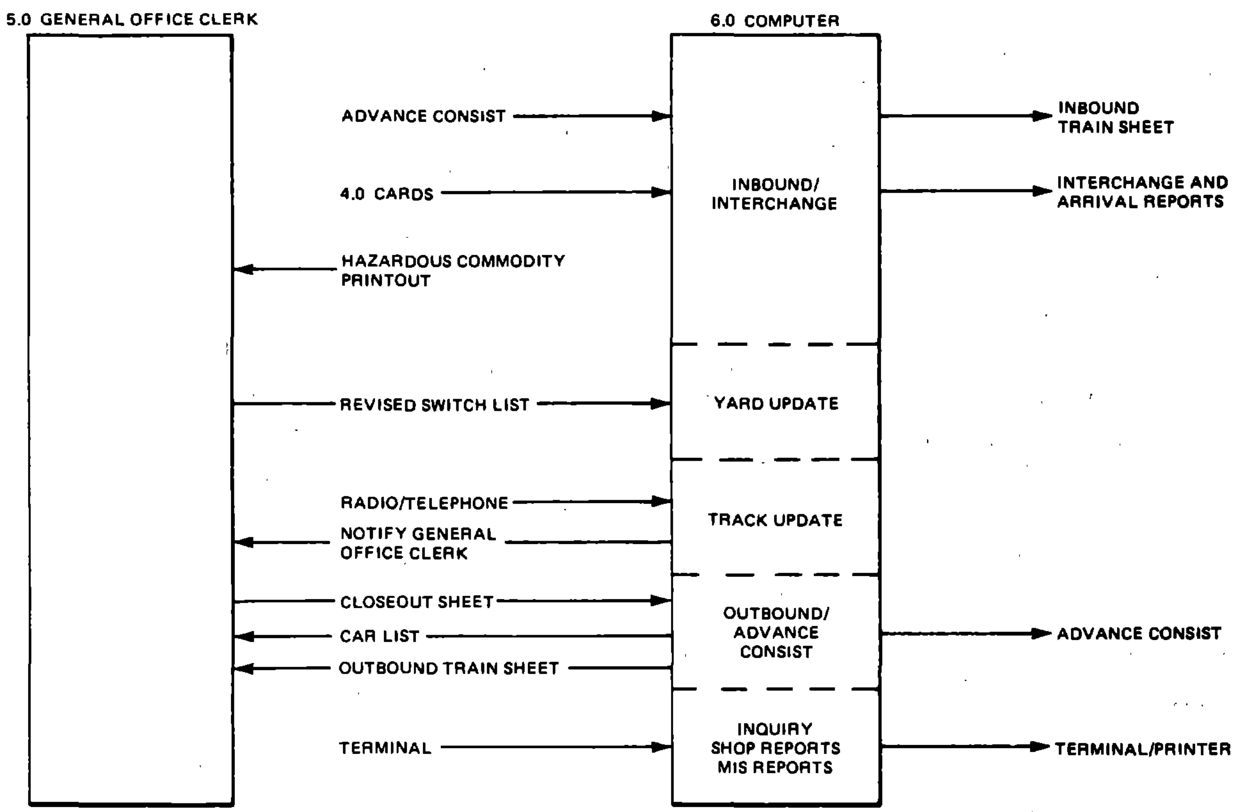
With the matrix structure, data can be gathered systematically, independent of interrelationships. Functional descriptions and numerical data are gathered on the current situation and are developed for one or more hypothetical alternatives from the feasibility study. If

	MIS	PC		COMPUTER FUNCTIONS
Function / Volume Quantifiers	Inventory Structure	Humping	Speed Control	Reliability Response Time
Number of cars	X			
Hump rate	X	X	X	
Report frequency	X			X
Volume of inquiries	X			X

FIGURE 5-6. SAMPLE MATRIX OF FUNCTIONS AND QUANTIFIERS FOR STRUCTURED ANALYSIS

alternative yard scenarios are of different sizes or traffic volumes, for example, quantitative values are scaled to reflect these differences.

Once quantitative data have been gathered, flowcharts can be used to document the current procedures in the yard. These represent the sequence in which a person or machine completes a series of functions. Each flowchart contains a number of boxes representing functions, with arrows between each function to represent the physical relationships and data flows between functions. Figures 5-7 and 5-5 present examples the current operations documented in the case study of Potomac Yard. For each person or machine, a more detailed description of the internal process must be documented. Exhibit 5-1 details the current



*Refer to Figure 5-5.

FIGURE 5-7. CURRENT COMPUTER PROGRAMS AT CASE STUDY YARD

EXHIBIT 5-1 CURRENT MIS PROGRAMS AT CASE STUDY YARD

<u>Inbound</u> (cars enter inventory)

- Reads RFPCOMFILE; one record per car, once per train
- Creates MASTERFILE (200 characters, 4/block)
- Builds CLINVFILE; contains ID, loaded/empty (L/E) status, weight, block number, and track number (11 characters, 45/block)
- Creates statistical record
- Produces inbound train sheet
- Interchange reporting to RFPCOMFILE

<u>Yard Update</u> (inventory change from humping)

- Inventory records moved to WORKFILE, deleted in CLINVFILE
- Cutoffs entered
- WORKFILE updates CLINVFILE, deleted in WORKFILE
- Move recorded in MASTERFILE
- NEWYARDM records movement in CARRECFILE

<u>Track Update</u> (inventory change)

- Records in-yard movement on CLINVFILE and MASTERFILE
- Reports shop movements to RSICFILE

<u>Outbound</u> (cars leave inventory)

- Updates STATISFILE
- Updates CARRECFILE
- Reads CLINVFILE to OUTFILE, deletes CLINVFILE
- Moves consist information to RFPCOMFILE or cards
- Prints outbound train sheet
- NCRINSET moves MASTERFILE to CARRECFILE

<u>Interchange</u> (create interchange report)

- Input of day/month
- MASTERFILE and CARREDFILE read to create ICFILE
- Information to tenant lines via RFPCOMFILE or cards
- Daily or monthly reports printed

MIS REPORTS

File	Frequency
STATISFILE	
Trains received and forwarded	Daily
Cars handled	Daily
In/outbound train statistics	Daily
In/outbound train statistics	Monthly
Statistics to Richmond	Monthly
FLAGFILE	
SCL load count	Monthly
B&O, CONRAIL tonnage	Monthly

INQUIRY

INQUIRY	Files Accessed	Sort
Car status		
Current	MASTERFILE and/or	By ID
All history	CARRECFILE	
Car status with waybill		
Current	MASTERFILE and/or	By ID
All history	CARRECFILE	

INQUIRY (concluded)

INQUIRY	Files Accessed	Sort
Return routing By line latest listing	MASTERFILE, CARRECFILE, 60DAYFILE	By ID and tenant line
Dangerous cars (display last hump, ID, track number, block number, L/E)	FLAGFILE CLINVFILE	Last event For all dangerous loads
Four-hour report--by line (display last hump, L/E for each block, total per line)	FLAGFILE CLINVFILE	Last event By tenant line and block number
Per diem (display ID, block number, track number, L/E of specific block of cars)	FLAGFILE CLINVFILE	Last event By block number
Track inventory--by yard and track (display last hump, L/E/wt totals)	FLAGFILE CLINVFILE	Last event By track number
Track inventory--by yard and track (display last hump, L/E/wt totals, listing of ID, LE, block, track)	FLAGFILE CLINVFILE	Last event By track number (read twice)
Track inventory--by yard (display last hump, number and weight totals, number and weight of cutoffs by block)	FLAGFILE CLINVFILE	Last event One yard
Track inventory--by track (display last hump, L/E/wt totals first and last car's ID, L/E, block, track)	FLAGFILE CLINVFILE	Last event By track number
Tonnage report--by outbound train (display last hump, number of cars track number, block number, and total weight for specific blocks and totals for train)	FLAGFILE CLINVFILE	Last event By block number

Potomac Yard MIS computer programs. Another example of a flowchart of yard operations is Figure 5-8. Exhibit 5-2 provides details for this example.

Functions in the yard can be grouped into subsystems of similar functions. An example from the Potomac Yard case study is the yard inventory subsystem. Figure 5-9 shows the relationships

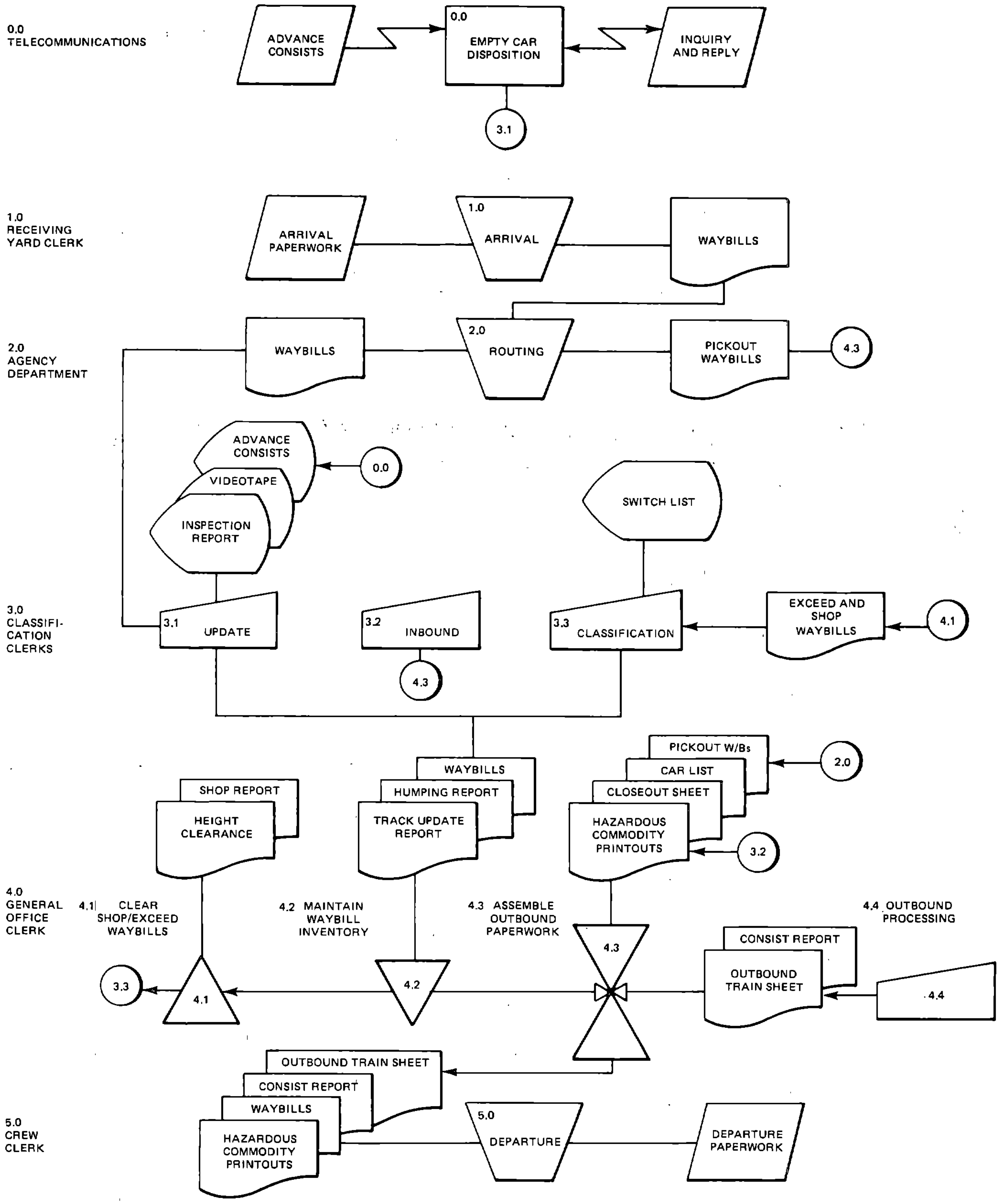


FIGURE 5-8. FLOWCHART OF FUTURE CLERICAL OPERATIONS AT CASE STUDY YARD

EXHIBIT 5-2 DETAILED PROCEDURES FOR NEW PAPERWORK PROCESSING AT CASE STUDY YARD

0.0 Telecommunications
- 0.0 Empty car disposition
  - In: Advance consists
    - Empty car disposition
  - Do: Create consist file
    - Update destination from empty car disposition
  - Out: Advance consist to 3.1.

1.0 Receiving Yard Clerk
- 1.0 Arrival
  - In: Train arrival paperwork
  - Do: Train arrives, route waybills to agency
  - Out: Waybills to 2.0

2.0 Agency Department
- 2.0 Routing
  - In: Waybills from 1.0
  - Do: Check routing, photocopy, and forward pickout requests
  - Out: Waybills to 3.1
    - Pickout waybills to 4.3

3.0 Classification Department
- 3.1 Update consist records
  - In: Waybills from 2.0
    - Videotape of arriving train
    - Inspection report
    - Advance consist
  - Do: Check consist against videotape
    - Update consist from waybills
    - Update consist class/status from inspection report
    - Update waybill from empty car disposition
  - Out: Waybills to 4.2
- 3.2 Inbound
  - Do: Enter arrival information to inbound train
  - Out: Hazardous commodity printout to 4.3
- 3.3 Classification
  - In: Preliminary classification guide
    - Exceed and shop waybills from 4.1
  - Do: Verify classification guide using CRT terminal
  - Out: Waybills to 4.2

4.0 General Office Clerk
- 4.1 Clear shop and exceed waybills
  - In: Shop car report
    - Height clearance
    - Waybills from 4.2
  - Do: Release shop and exceed height waybills for classification
  - Out: Waybills to 3.3
- 4.2 Maintain Waybill Inventory
  - In: Waybills from 3.1 and 3.3
    - Track update report printout
    - Humping report printout
  - Do: Waybills are sorted and filed in track-standing order
    - Waybills are pulled for classification and departure
  - Out: Shop and exceed waybills to 4.1
    - Departure waybills to 4.3
- 4.3 Assemble Outbound Paperwork
  - In: Closeout sheet from trainmaster
    - Car list printout
    - Hazardous commodity printout from 3.2
    - Waybills from 4.2
    - Pickout waybills from 2.0
    - Consist report printout from 4.4
  - Do: Pull departing waybills
    - Assemble waybills and hazardous commodity printouts, outbound train sheet, and consist reports
  - Out: Assemble paperwork to 5.0
- 4.4 Outbound Processing
  - Do: Prepare consist report and outbound train sheet using CRT terminal
  - Out: Outbound train sheet printout to 4.3
    - Consist report printout to 4.3

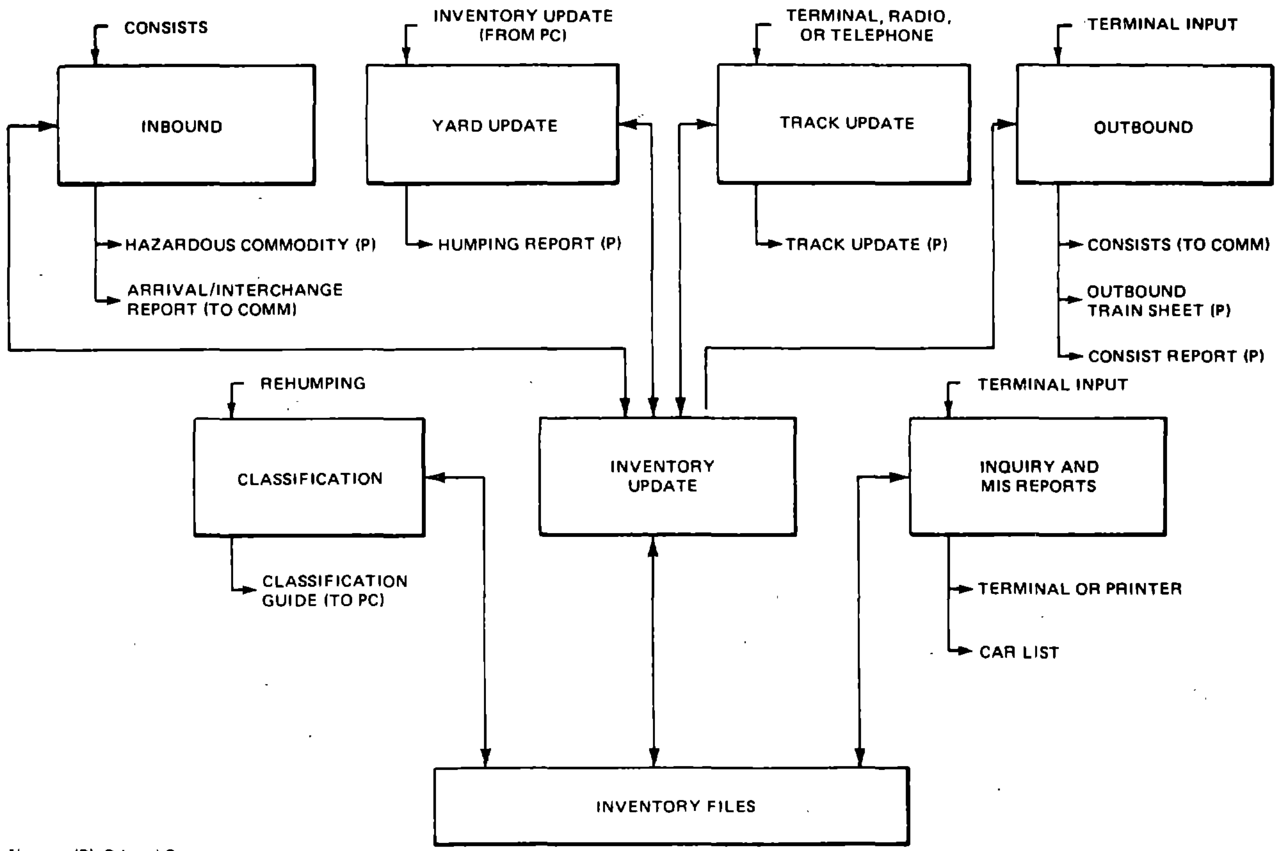
5.0 Crew Clerk
- 5.0 Departure
  - In: Outbound train sheet from 4.3
    - Consist report from 4.3
    - Waybills from 4.3
    - Hazardous commodity printouts from 4.3
  - Do: Log information in outbound log book
  - Out: Paperwork to departing crew

---

*Refer to Figure 5-8.

between various functions. Figure 5-10 shows in more detail the input and output and relationship between MIS functions, programs, and files. An overview can be drawn, as in Figure 5-11, to show the interrelationships of subsystems within the total system. For each flowchart, detailed information gathered [software programs, file sizes, and information flow (data dictionary), and the frequency of each function] should be identified and listed in a separate document. The logical processes of each function and the format of all input (communication, keyed entries, cards) and all output (reports, screens for inquiries) should also be documented.

5.2.1.2 Current Level of Automation. A complete survey should be conducted of the current hardware, software, and communications equipment used in the yard and any hardware with which new yard hardware would interface. This list should include for each item of hardware a complete description, original cost, expected time of replacement, replacement cost, and space, power, and environmental requirements. EDP personnel should be contacted about which hardware components are irreplaceable because of custom design. Figure 5-12 is an example of a system configuration from the case study.



Note: (P), Printed Output
(PC), Process Control System
COMM, Communications Subsystem

FIGURE 5-9. YARD INVENTORY SUBSYSTEM AT CASE STUDY YARD

5.2.1.3 Physical Attributes of the Yard. A complete study of the yard must include data on traffic, yard layout, and current field equipment. Traffic data should comprise arrival schedules and mix of cars, number, type, and frequency of moves within the yard, and departure schedules and mix of cars. For the yard layout description, the project team should prepare a track diagram, specify track capacity, and formulate a detailed list of all field hardware. Current humping rates and crew schedules should be computed.

5.2.2 Develop Functional Model of the New System

The first step in modeling the new system is to define a logical (functional) equivalent to the current physical system. The actual tasks are generalized to represent the underlying goal. To represent the logical equivalent of the physical data flow, only functions of the yard and their interfaces are represented using a data flow diagram. The representation is independent of physical hardware. Figure 5-13 is a simple data flow diagram from the case study.

The abstract representation of the current system is then used to develop a similar representation of the new system. Requirements and goals of the new system are added as functions of the new logical model. This new model must meet the charge of management made in the feasibility study. Figure 5-14 depicts the functional (logical) model of the upgraded yard information and control systems in the case study. No assumptions of hardware are made when developing the logical model. Figure 5-11 presents the functional grouping (partition) used in the case study to design hardware alternatives. The logical interface between subsystems indicate how the physical systems will communicate.

In the structured analysis method, a complete description of each logical system includes data flow diagrams and a data dictionary. The data dictionary lists all files and the specific data required at each interface and describes the process involved in each step in the data flow diagram. This description is the basis for the eventual functional specification.

In many cases, one functional model may not be adequate because the level of sophistication and scope of functions may not yet be well defined. The feasibility study may have indicated a number of options to be analyzed. The choice between options is made as more comparative information is available after the cost-benefit study (Step 5).

To fully develop alternatives, the project team must define the functional requirements for the

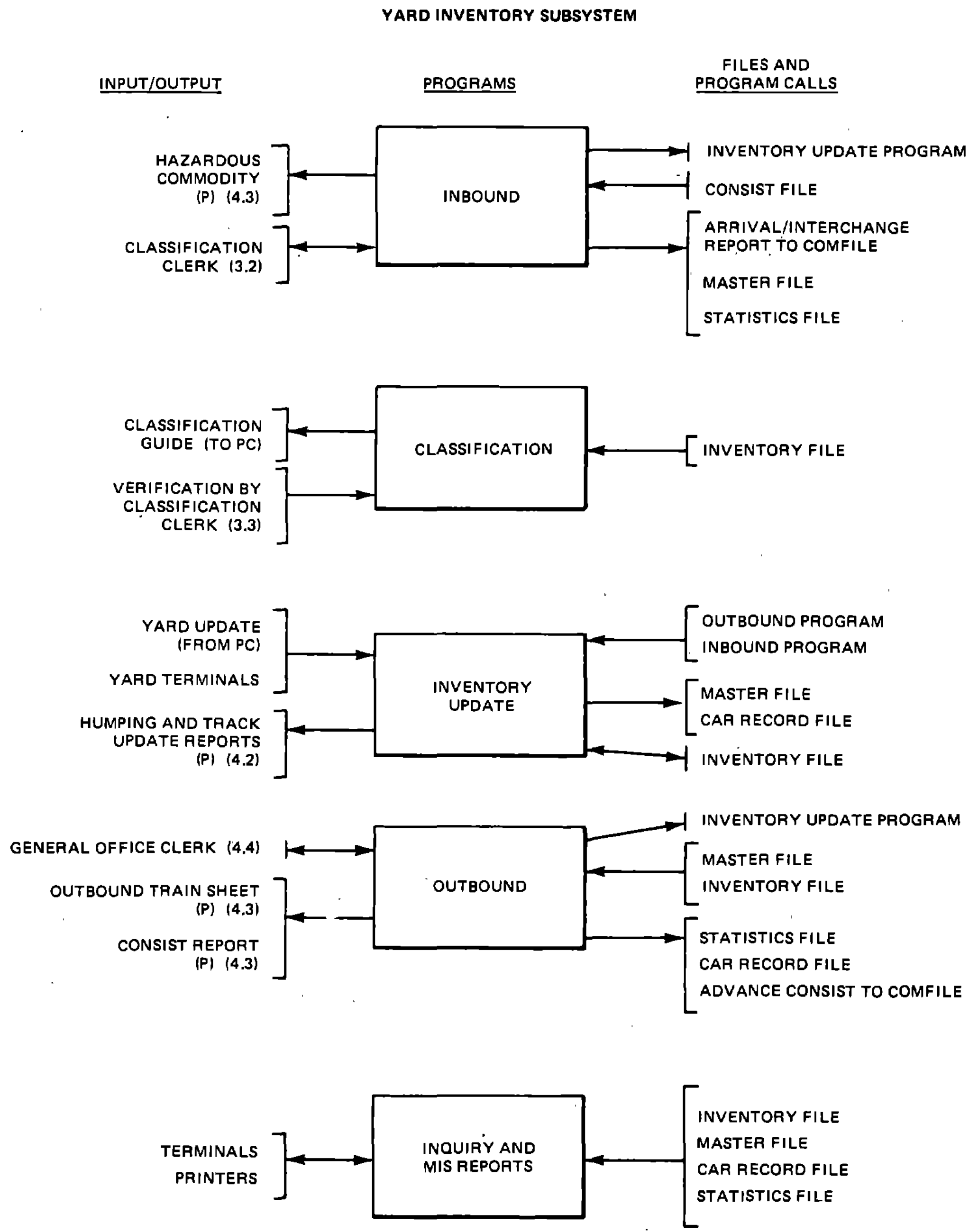
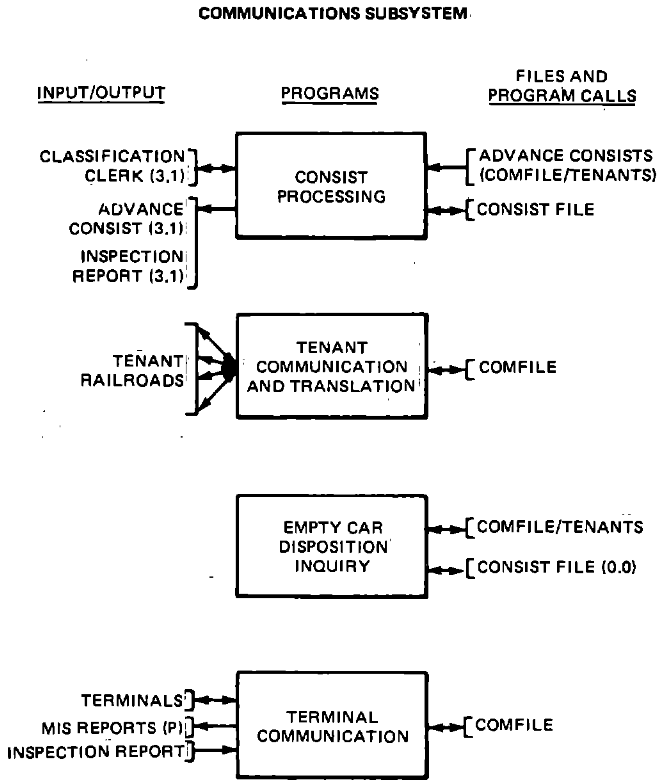


FIGURE 5-10. NEW MANAGEMENT INFORMATION SYSTEM PROGRAMS AT CASE STUDY YARD



Note: (P) = Printed Output; (PC) = Process Control System; all numbers in ( ) refer to Figure 5-8.

FIGURE 5-10. (Concluded)

upgraded yard. General guidelines should be a by-product of the original feasibility study. If further clarification is required, the project team members must interview both users and management. This approach, however, most likely will produce ambiguous results requiring a number of different functional alternatives. Described below are some alternative MIS functions.

The purpose of the yard MIS is to provide better and more timely information so as to enhance operational efficiency. Its function can be subdivided into two groups: (1) those that primarily benefit system central data processing and (2) those that benefit the local yard. However, policies and procedures regarding the division of responsibility between central and local differ from one railroad to another; a clear-cut grouping scheme is therefore not possible. Exhibit 5-3 depicts a division into systemwide MIS, yard MIS, and PC functions. Discussions in the design manual should be limited to local yard functions and their relationship to central systemwide functions. Section 5.2.3 discusses the distribution of functions between central and yard locations.

5.2.2.1 Communication. The communications function is to take information from yard computers and transmit it to the systemwide MIS computer. Some yards, such as Potomac Yard, may have more than one systemwide MIS computer. At Potomac Yard, the MIS computer must communicate with the computers of six different railroads.

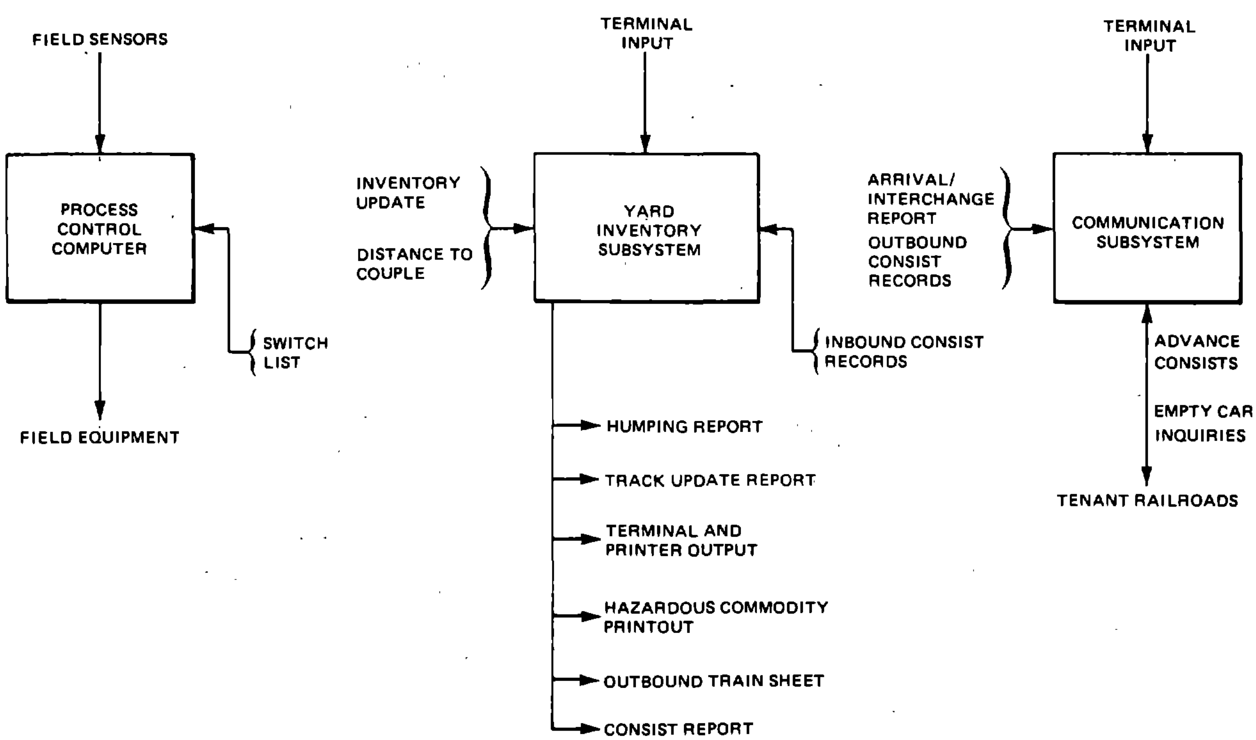


FIGURE 5-11. SYSTEM INTERFACES

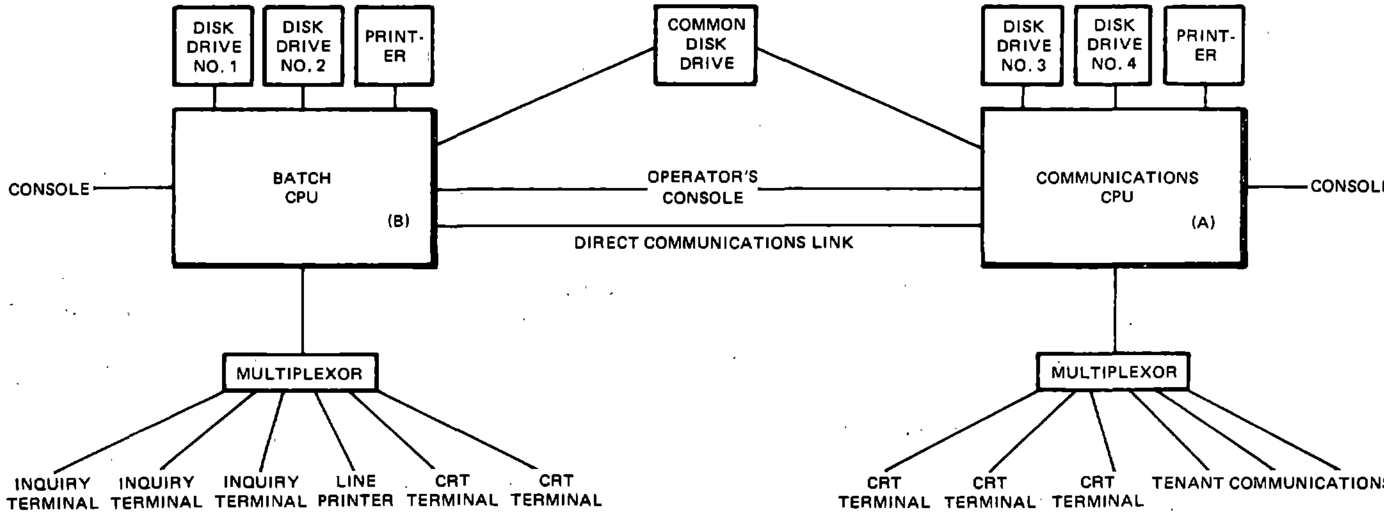


FIGURE 5-12. SAMPLE HARDWARE CONFIGURATION FROM CASE STUDY YARD

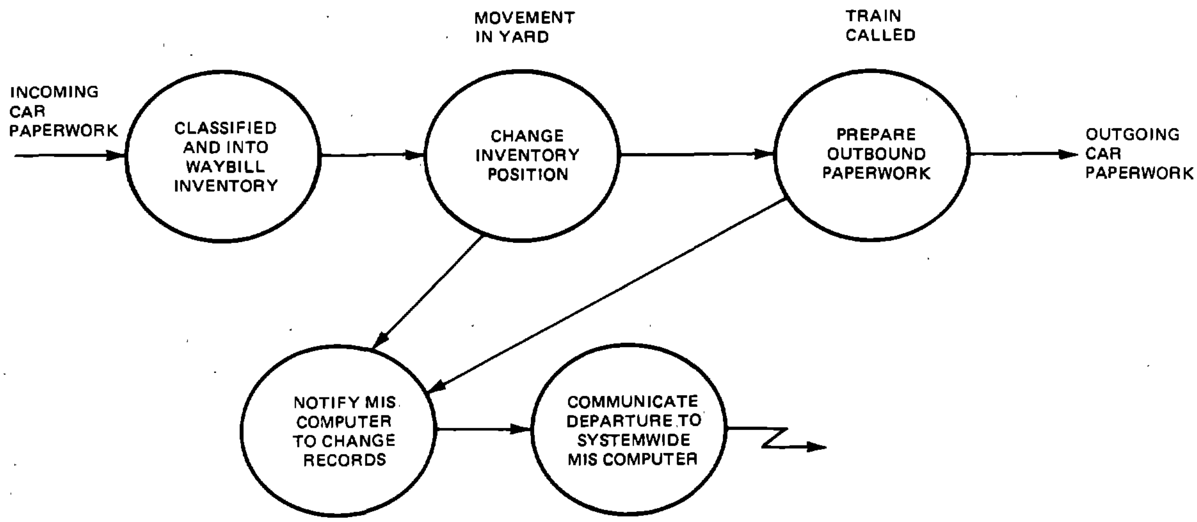


FIGURE 5-13. SAMPLE DATA FLOW DIAGRAM OF CURRENT OPERATIONS FROM CASE STUDY YARD

The method of telecommunication must also be established. Existing communications may be by telephone or by teletype, but new functions may require direct communications or communications to an intelligent terminal that keypunches the required information onto punched cards. Systemwide reporting requirements for consists, inquiries, and interchange reports are used to estimate the volume of messages and the frequency and size of the messages. Sometimes, the two communications computers will not use the same format, symbols, or abbreviations. In that case, the communications function must also translate some of the messages, which requires both additional processing power and the storage of translation tables. The capacity of the communications used and the volume sent dictate the amount of buffer memory required.

5.2.2.2 Yard Status and Car Records. The project team must determine whether transmission of arrival and departure reports to the systemwide computer or other yards is required. The number of reports will correspond to the expected number of trains arriving and departing.

If advance consists must be sent and received, the expected volume must be estimated. If consist information is often inaccurate, it cannot be used for the planning of yard operations until it is verified by the arriving train. The size of the consist record for each car influences the communications load.

Through inbound processing, the arrival of a train is recorded, and its individual cars are entered into the inventory. To design the program, the project team must study current inbound processing procedures and decide how they should be done in an upgraded yard. The project team must consider: whether car lists in a specific order are required for inbound inspections;

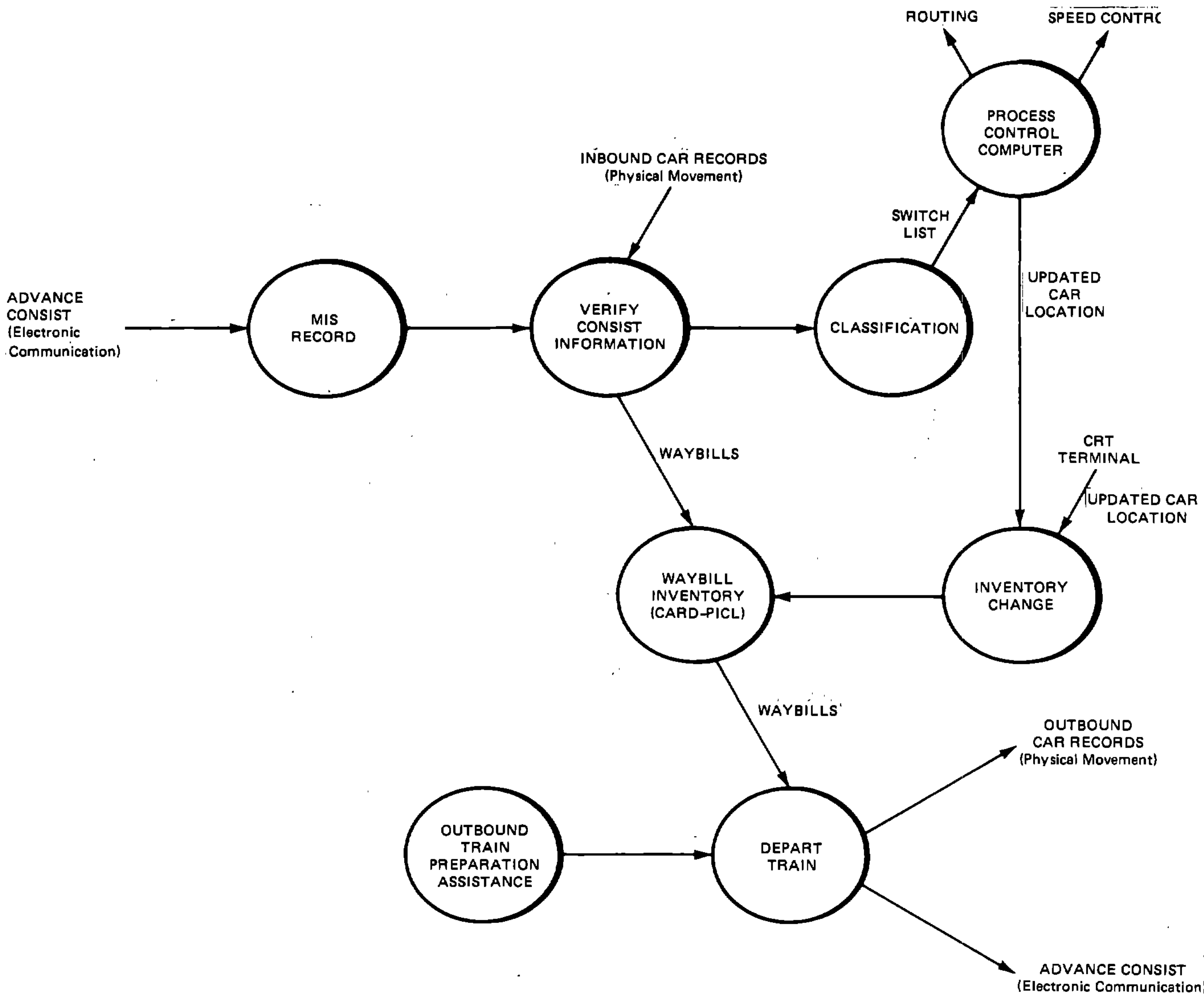


FIGURE 5-14. SAMPLE DATA FLOW DIAGRAM OF FUTURE OPERATIONS FROM CASE STUDY YARD

how the completed inspection results are reported; whether the results are to be entered onto terminals placed in the field; and how soon all inbound processing must be completed so as not to delay switch list generation for humping. The longer car records remain as part of advance consist or inbound records before becoming inventory records, the larger the file needed.

If waybills are kept as computer records, the project team must determine how they are entered; whether they arrive as part of the advance consist, and whether a repetitive waybilling program is needed; if so, the team must assess whether waybills will add a load to the printing required or will require special higher quality printing.

Outbound processing removes departing cars from inventory and builds the departure and advance consist reports. Whether the program also prints inspection lists, monitors departure preparation, and estimates engine requirements depends on the present and expected operating procedures. The project team must decide whether data entry terminals will also be required in the departure yard and identify when the outbound processing becomes critical (e.g., when slow processing might delay a departing train).

5.2.2.3 Inventory. The size and structure of the inventory files depends largely on the size of the yard and the size and number of inventory reports and inquiries required. (The relationship of reports and inquiries to file structure was discussed in Chapter 3.) The size of the inventory is determined by the maximum expected number of cars that might ever be in the yard. The size of the record for each car also influences the file size. A record containing only car IDs will create a much smaller file than will car records containing destination, load, and status codes. The actual record size is determined by the report requirements and how frequently the reports are needed. The number

EXHIBIT 5-3 FUNCTIONAL HIERARCHY

Note: C denotes centralized, and
D denotes decentralized.

SYSTEMWIDE MIS

Inventory by yard, detailed car information
Detailed waybill information/routing
Demurrage/interchange records
Empty car disposition
Accounting

Interyard communications (C)
Paperless record exchange for waybills and interchanges (C)
Advance consists (C)
Agency system
Motor power unit disposition

YARD MIS SYSTEM

Detailed car information, track number, consist information, and history, by car ID
Inventory of industrial tracks
Yard history by train arrival/departure
Clerical handling of waybills
Routing by waybill (local cars)
Routing by waybill (D)
MIS reporting
Advance consist to/from systemwide MIS (C)

Determine humping sequence of trains
Expected cars by block to be humped
Expected train departure
Interyard communications (D)
Paperless record exchange for waybills and interchanges (D)
Advance consist to/from other yards (D)
Accounting (D)
Resource scheduling
Dynamic track assignment

YARD INVENTORY SUBSYSTEM
(Part of either yard MIS or PC system)

Inventory by car position in track; track car movement in yard
Switch list generation
Classification table
Swing tracks by block

Inventory reports/inquiry
Outbound car list

PROCESS CONTROL SYSTEM

Physical control of cars
Speed control
Distance to couple—track occupancy
Automatic routing and switch control
Return updated switch list
Trim-end switches

Reports
Alarms--height, cornering, catch up, lockouts, hot-box detection
Hump engine speed control
Manual control override

of tracks must be known, as well as the length of each track and the maximum track occupancy. The project team must decide whether car movements and inventory corrections will be identified by car ID or just by the track position; whether the inventory will include all receiving, departure, classification, and shop tracks; and whether yard engines will be included in the inventory. The information required for cars in shop tracks is determined by the reports required.

5.2.2.4 Switch List Generation. Switch lists are created by matching the block or destination of each car with an assigned track in the classification yard. This function requires processing time, memory space to hold the classification table, and a communications medium to transmit the results to the PC computer. To design the switch generation function, the number of trains and cars humped each day and their block distribution must be known. The complexity of the classification is determined by the number of cars humped, the number of blocks and tracks, and the car distribution. This will also indicate the memory required for the classification table.

The project team must also determine whether permanent or temporary changes will occur often. If so, a program must be designed to accommodate change of the classification table when necessary. If the PC computer is to update inventory as each car clears, communications will be more frequent than if an updated switch list is returned after each group is humped or if the exceptions to the original switch list are returned.

If a number of rehumpings occur each day, they must be manually entered or if all the rehumped cars are pulled from one track, the track inventory can be used for the switch list generation. The operation of the hump determines the criticality of the switch list generation and the required availability and allowable downtime of the MIS computer.

5.2.2.5 Reports and Inquiry. The type of report determines the size and type of file management system required, and the frequency of issuance of these reports and inquiries also influences the processing requirement. The reports and inquiries also determine the number, size, and locations of printers and inquiry terminals. Reports and inquiries might be limited to specific terminals or individuals. The frequency of issuing a report will be less if the report is only produced on an exception basis--printed only when the information is out of the ordinary. Any report that need not be processed immediately is queued to be printed later. In this manner, the computer can process it during a slack period.

5.2.2.6 Other. The project team must also consider the number and location of terminals and printers. For long distances (greater than 2,000 feet), terminals require boosters to receive signals from the computer. If access to the computer or specific programs and information is restricted, the project team must document these details. If the yard computer is to be used for additional functions, such as engine and crew monitoring, management decision aids, and accounting, the project team must also design those functions to estimate their effect on the MIS functions.

5.2.3 Develop Functional Configurations

The goal of systems design is to determine the best configuration for the functions defined in the logical model. Many functions in the logical model of the yard will require automation. The automated functions may be part of one large computer or they may be divided among a number of smaller systems.

Step 4 (Figure 5-4) is to designate which functional processes will be automated and which will remain clerical/manual tasks. Alternative hardware configurations can be developed in a systematic manner by exploring the distribution of functions among independent units of hardware. Elements that constitute the functional design of the proposed system can be used as building blocks within the configuration of hardware and software components. For each alternative, each functional interface in the logical flow diagram will be internal to a computer or a person/department or will be a computer-to-computer or man-to-machine (computer) interface. Data flow diagrams depicting required functions can be used to guide the design of alternative hardware configurations. The relationship between two different functions can be analyzed by noting the volume, type, and frequency of information transferred, files that are shared, input and output devices that are required, and people required for each function. Dependent functions are more likely to be grouped within a subsystem or software module. Figure 5-15 is an example of a data flow diagram showing such groupings.

Computer functions are distributed among computer systems: systemwide MIS, yard MIS, and PC computer systems (see Exhibit 5-3). The distribution of functions between the railroad central and the yard is often determined by present EDP resources and data processing management philosophy. Once the location of functions is determined, functions must be allocated among units of hardware. Determining factors include cost, functional specifications, and redundancy requirements. Functional distribution can be structured along a number of dimensions including: location in the yard, common processing functions (communication, I/O, data entry, file manipulation), application (real time control, car inventory), and location within the yard organization (e.g., clerical, engineering, signals).

The systemwide MIS computer is located at systems central. Among the possible applications are: systemwide inventory; processing of interyard and interrailway consists, waybills, and financial information interchanges; central traffic control; and planning and simulation. Union Pacific's Complete Operating Information (COIN) system, which tracks car and train movements, services inquiries and produces records on movement history, current movement, and the yard location of cars. Other examples are Southern Pacific's Total Operations Processing System (TOPS), Missouri Pacific's (MOPAC's) Transportation Control System (TCS), and Canadian National Railway's TRACS.

Systemwide MIS computers usually do not provide a track-standing inventory within specific yards. This function is provided by the yard MIS computer. A yard MIS computer usually: is located at the yard, provides processing only for that yard, and is separate from but in communication with the PC computer. MOPAC's Yards and Terminal System (YATS) is an example. YATS uses a computer system in the yard to maintain a computerized car inventory, support local management information requirements, generate car classification work orders, permit data entry via CRT terminals, and provide a real-time, on-line data base for local operations analysis (Digital Equipment Corp., 1975). In addition, each YATS site can be used to access the TCS, a centrally located systemwide MIS computer.

Another example of a yard MIS system is Union Pacific's Terminal Information Systems (TIS). A TIS computer communicates with a yard PC computer

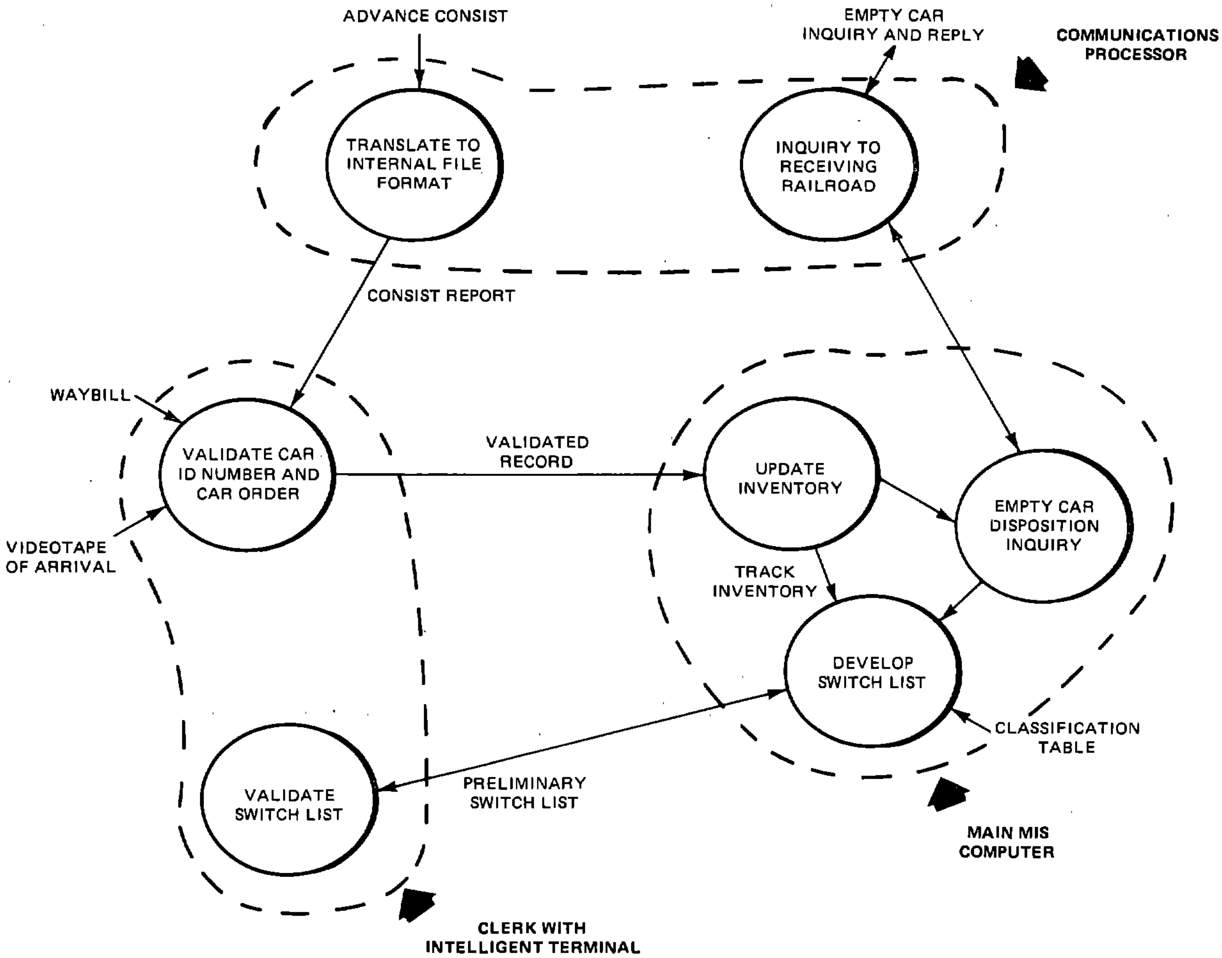


FIGURE 5-15. EXAMPLE OF DIVISION OF FUNCTIONS TO DEVELOP A CONFIGURATION

to transmit switch lists and receive reports of inventory movement. TIS differs from YATS in that it has the capability to act as the yard MIS computer in a number of yards. TISs are to be located in major yards and can be expanded to handle additional smaller or satellite yards, with each yard having a separate data base and the same local data entry/inquiry capabilities.

5.2.3.1. Geographic Distribution of MIS Hardware. MIS hardware configurations may be centralized, decentralized, or regionalized. In a centralized configuration, the yard MIS computer is located at systems central with the systemwide MIS computer. Individual yards do not have on-site computers but communicate with central computers via communications lines and I/O terminals. Figure 5-16 presents a typical centralized system. Yard I/O terminals are typically CRT terminals and teletypes, as in Southern Pacific's West Colton Yard. At West Colton, the yard (TCC) and systemwide (TOPS) MIS computers are located in San Francisco. Microwave communication links connect to CRT terminals, teletypes, and printers in the yard.

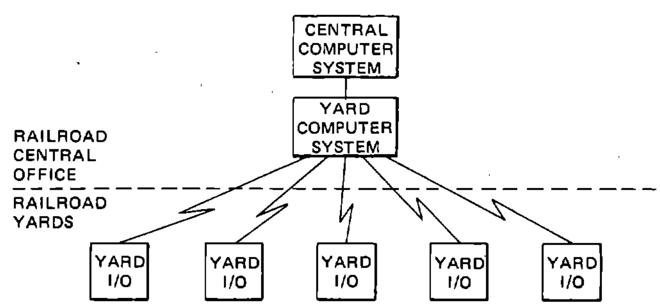


FIGURE 5-16. CONFIGURATION OF A CENTRALIZED YARD COMPUTER SYSTEM

MOPAC's Yard Support System is another example of a centralized system. MOPAC has independent yard computers in its largest yards but uses programmable communications terminals in its smaller yards (Lynch, 1977). These terminals store and process a small inventory but are primarily data entry devices for the systemwide TCS.

Centralized control need not be limited to hardware configurations. Standardized hardware and software packages and operating procedures in yards indicate centralized control. The major factors that suggest a centralized system is preferred are the rail network characteristics (a linear rather than grid system), a homogeneity of applications, the low cost of communication lines, and the labor union rules governing staffing in local yards.

Under a decentralized approach such as the one depicted in Figure 5-17, the yard MIS computer is installed at the individual yard and is under the control of local management. Examples of systems where independent yard computers are remotely located are MOPAC's TCS/YATS and TCS/Incoterm systems and Family Lines Rice and Strawberry yards.

CONRAIL is decentralized with some modification. One inventory is kept at each decentralized local yard, and a separate inventory is kept for each yard at a central location. The yard-resident inventory file structure is optimized to produce a track list, while the centrally located yard inventory is designed to optimize the search for individual cars.

Although the computers are physically independent, links to a central system require the existence of some central control in hardware and software. Usually, data input and report writing software are customized for individual yard needs.

The regional approach may be viewed as a compromise between the centralized and decentralized approaches. Under this system, yards are grouped into regions, with each region having its own MIS computer. The regional yard computer may reside at the system central or at a large classification yard. Most yards have only I/O terminals connected on-line to the regional computer. Figure 5-18 presents a typical regionalized system configuration.

Grand Trunk Western Railroad uses regional computers to collect data from and send processed information to on-line terminals located in classification yards (Lay, 1980). This configuration concentrates the flow of data input and transmission to the central headquarters' computer complex. The geographic coverage of regional centers is determined by operating territories and traffic volume.

Union Pacific is planning to install computers for its TIS system in 13 major yards. These will be independent yard computers linked to the central systemwide COIN system in a decentralized configuration. When expanded, these computers will act as regional centers handling the processing and communications of smaller satellite yards.

Management philosophy and the structure of existing computer resources are the greatest determinants of MIS network architecture. (This is discussed in more detail in Section 5.2.5.) In the installation of a new system or modification of an existing system, the most difficult design problem is the distribution of functions among the hardware at each site.

5.2.3.2 Functional Distribution of Yard Hardware. Yard MIS and PC functions may be allocated to local hardware using a variety of approaches. For example:

- MIS and PC functions are performed by the same computer.
- The MIS computer and PC computer are separate and are linked through off-line data communication only.
- The MIS computer and PC computer are parallel linked with a larger central computer performing railroad system MIS functions. An on-line linkage also exists between the two at the local level.
- MIS functions are divided between an MIS or communications computer and the PC computer. The yard inventory is kept by the PC computer.

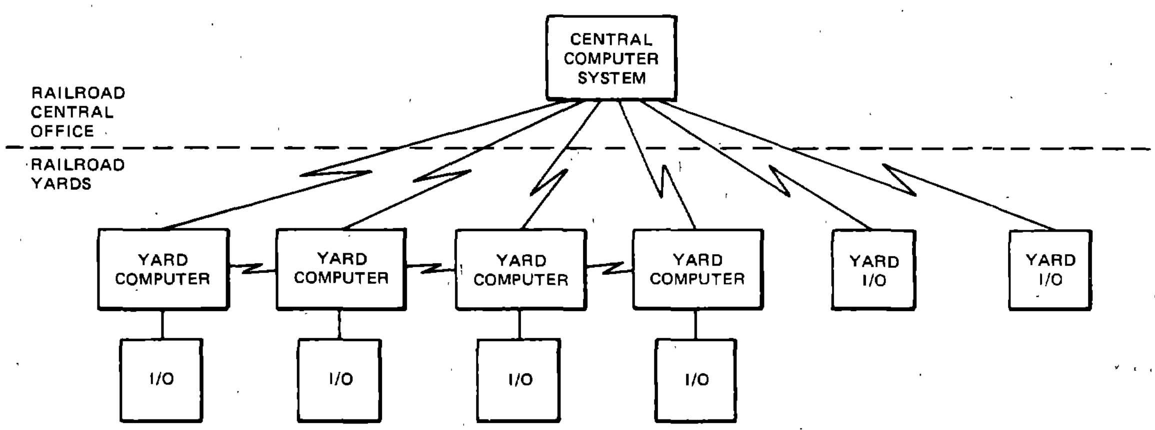


FIGURE 5-17. CONFIGURATION OF A DECENTRALIZED YARD COMPUTER SYSTEM

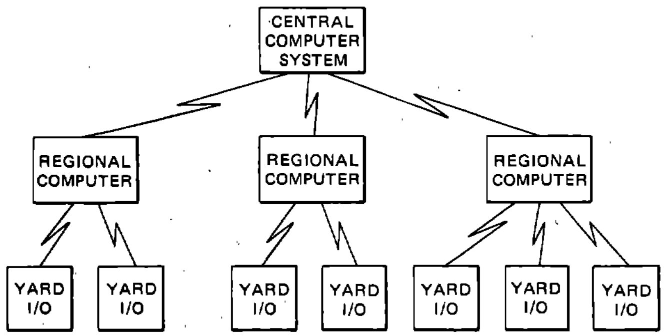


FIGURE 5-18. CONFIGURATION OF A REGIONALIZED YARD COMPUTER SYSTEM

- The PC computer is subordinate to the MIS computer.
- The MIS computer is subordinate to the PC computer.

In addition, an MIS, PC, or combination MIS/PC computer may actually consist of a number of smaller computers. An additional computer may act as a backup to the primary computer. With distributed processing, functions are distributed equally among multiple computers that link to each other to share the processing burden.

Santa Fe's Argentine Yard is an example of a PC computer with backup. MIS and PC functions are separated, with a computer in the terminal office building handling MIS functions and two computers in the hump yardmaster's office in charge of process control. One of these PC computers directs PC operations and the other is a backup.

Two yards that have one computer performing both PC and MIS functions are Canadian Pacific's Alyth Yard and Southern Pacific's Eugene Yard. At Alyth, two computers--one primary and one backup--provide automatic retarding, routing, and hump engine control and produce operational reports and car inventory. Eugene Yard has one computer for PC and MIS functions, with a relay backup. One possible disadvantage to having the same computer perform PC and MIS functions is that MIS functions may interfere with time-critical PC functions.

An alternative to having a single computer perform all PC functions is to distribute the various control functions among several minicomputers. Two of the advantages of this system are being able to operate at least part of the yard automatically, even if one of the computers breaks down, and being able to upgrade an old yard gradually, since the system is modular (GRS, 1978). For example, a minicomputer can be installed to control the retarding, while switching can still be performed by pushbutton. In an integrated system, control may be partial (e.g., control of only retarding, or control of only switching), but all control elements are linked to each other or to a central computer so that information may be passed between them. Southern Railway's Sheffield Yard in Sheffield, Alabama, has a distrubted system with five minicomputers in the yard:

- MIS
  - One inventory control minicomputer
  - One backup minicomputer (hot standby).
- PC system
  - One minicomputer for retarding and hump engine control.
  - One minicomputer for automatic route switching.
  - One backup minicomputer (hot standby).

These are supported by a central computer 235 miles away in Atlanta. When a train arrives in Sheffield, the central computer transmits car and train data and a classification code for each car to the MIS minicomputer. The MIS transmits various information, including a classification track assignment for each car and a switch list to the PC system (Martin & Durand, 1974). Another example is CONRAIL's Elkhart Yard, where five separate processors are used for automatic switching, automatic retarder control, PC backup, MIS applications, and MIS backup.

One variation of distributed process control is a partially automatic system in which mini- or micro-computers or microprocessors perform very specialized tasks in the yard without communicating with each other. Another variation is where PC functions are distributed among several data linked but not control-linked computer systems. For example, the use of a microcomputer at the pullout or trim end and a larger PC computer at the hump end.

5.2.3.3 Case Study Example. A simple example comes from the Potomac Yard case study. Figure 5-11 depicts the major subsystems involved and interfaces between them, and Figure 5-19 shows the three configurations that can be developed from the major subsystems. The current Potomac Yard configuration is such that all MIS functions are in one processing system. One alternative configuration is to add a second processing unit for PC functions. A second alternative configuration is to combine the PC and inventory functions in one unit of hardware, with the remaining MIS functions residing in an independent computer system. The third alternative is to use a single PC/MIS computer for all yard functions.

Each unit of computer hardware, although designated to a specific limited number of functions, may be structured in a number of ways. All the designated functions usually reside within one computer. One alternative is to use a number of systems to perform the same tasks to distribute the processing load and minimize the loss of any single processor. Another configuration is to use distributed processing, whereby different computers are used for the specific functions they do best. An example would be the use of smaller microprocessors to control retarder speed. In this case, the main PC computer would give a desired exist speed to

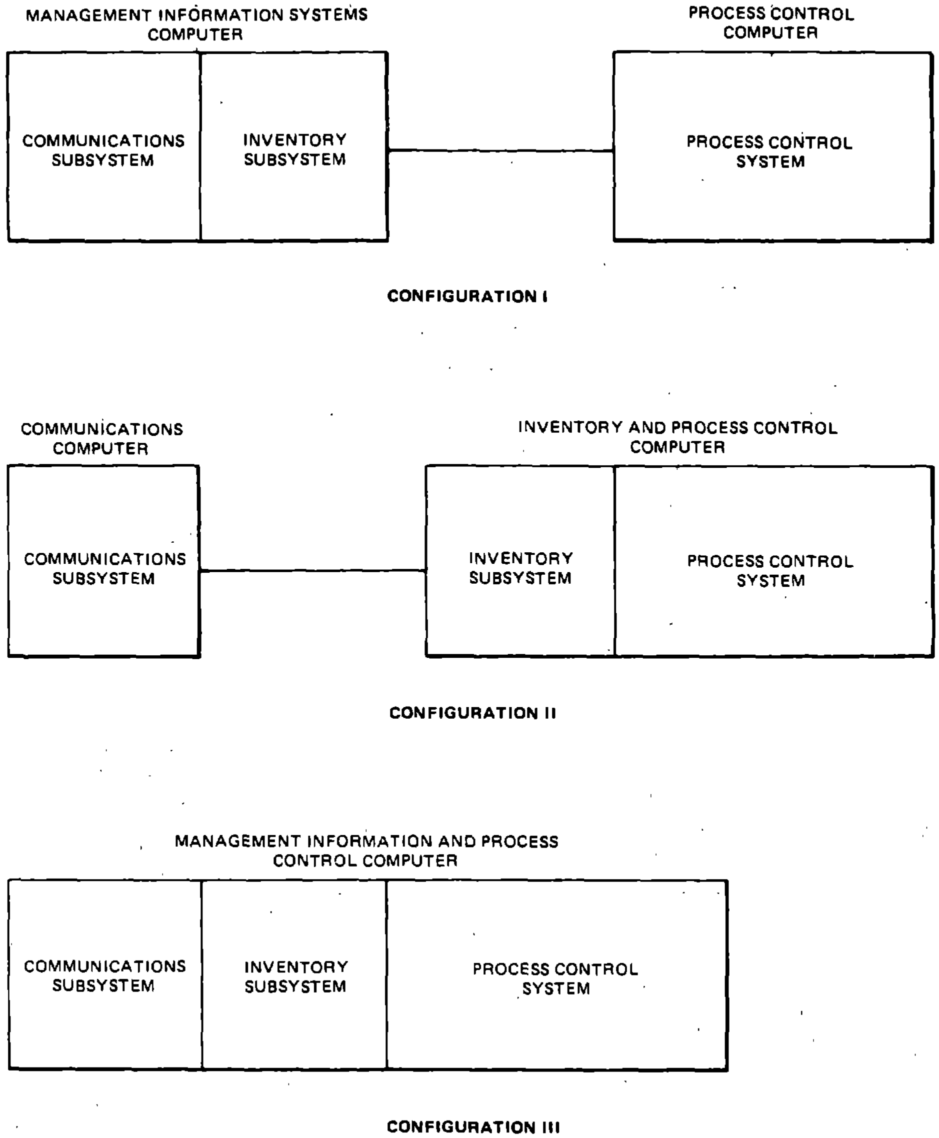


FIGURE 5-19. FUNCTIONAL ALLOCATIONS OF EACH ALTERNATIVE CONFIGURATION FOR CASE STUDY YARD

the microcomputer, the internal logic of which would control humping to provide smooth deceleration.

Another example is the use of a smaller front-end processor to provide communications control and queuing for the MIS computer. In the design of a distributed system, independent functions must be found that require few communications back to the main CPU. This allows the detached processors to work independently to minimize interruptions that hinder system efficiency.

5.2.4 Measure Costs and Benefits

The fifth step of the analysis phase is to measure the costs and benefits for each option. These data become one basis for selecting the best system design. Although only relative costs and benefits are required for the comparison of alternatives, a complete cost/benefit analysis is necessary for a comprehensive evaluation of the feasibility of the proposed project.

A number of alternatives will have been developed, and a cost/benefit study is one method to quantify each alternative. Important cost factors are how long yard services will be disrupted during the project and the delay until benefits are returned from the investment.

A gross estimate of hardware requirements for each alternative is necessary to estimate costs. For a PC system, many railroads have a vendor develop the hardware and software or install a predeveloped package. Informal costs estimates may be obtainable from vendors. For a more detailed analysis, the hardware study should be begun (see Section 5.4).

5.2.4.1 Cost Analysis Methods. The purpose of the cost/benefit analysis is to estimate all costs and benefits over the life of the computer system. Life cycle costing is then used to find equivalent monetary values with which to compare different alternatives that have varying equipment and manpower costs incurred at different times. Future costs can be discounted to an equivalent present value, annual cost, or benefit.

The present-value method assumes that each year of expenditure delay has a positive value equal to the interest that could be earned. Because of the positive value of delay, any future costs (future value, FV) may be discounted by that interest to find the present value (PV). The formula for converting future value to present value is:

$$PV = FV/(1 + i) \text{ for a single year}$$

$$PV = FV/(1 + i)^n \text{ for n years.}$$

To determine the total cost for any alternative, costs for each year must be estimated. For recurring costs over a number of years, the following formula is used:

$$PV = A \times \frac{(1 + i)^n - 1}{i(1 + i)^n}$$

where

PV = present value at beginning of series
A = uniform annual cost
i = discount (interest) rate
n = number of years.

For example, this formula can be used to compute the present value of yearly maintenance for hardware.

The total present value equivalent for an alternative is found by totaling the present value of each year's expenditures. The total is therefore:

$$PV = \sum_{j=0}^{n} \frac{FV(j)}{(1 + i)^j}$$

where

FV(j) = the cost incurred in year j
i = discount rate
n = years of the life cycle
PV = present value.

For each alternative, cost estimates must be made for equipment, installation, software development, and annual operating costs. Exhibit 5-4 is a sample work sheet for estimating capital and recurring costs. The cost, the year the cost is incurred, the life expectancy, and the salvage value are required for each capital expense item. For recurring costs, the annual cost and number of years incurred are required. When comparing different alternatives of different life cycles, a standard length of time for the study must be set.

5.2.4.2 Benefit Analysis. Benefits are much more difficult to identify than costs and very difficult to quantify accurately. Because a monetary comparison is being made, a monetary value of benefits must be attached when possible. Some benefits are easily identified and quantified. For example, if any equipment or personnel would be eliminated if a particular alternative were selected, this saving can be considered a benefit.

The following possible benefits can be identified with the installation of computer assistance:

- Lower clerical costs
  - Reduction in per-unit processing cost
  - Space saving
  - Reductions of redundant operations
  - Elimination of tedious, routine jobs.
- Improved accuracy of data
  - Elimination of clerical steps
  - Built-in data error checks.
- Better recordkeeping
  - Automatic data collection
  - Reduction in duplicate files.
- Faster reaction time
  - Information is more up to date
  - Easy access to timely information
  - Faster response to problems.
- Improved information
  - More detailed information available for management decisions.
  - Problems can be detected and corrected early.
  - Less important information can be filtered out.

Tables 5-1 and 5-2 identify some specific benefits derived from the use of MIS and PC computers. Although quantifiable benefits are gained from improved clerical operations, of greater importance is the improvement in yard operations and service. The difficulty is to estimate the amount saved as a result of better decisions and operational improvements. To quantify these benefits, actual performance improvements in yard operations must be estimated and the monetary benefits estimated from that. This is a difficult, inaccurate task. Some examples of monetary benefits are as follows:

- Meeting schedules accurately--Cars are better utilized; increased business may result from improved customer service.

EXHIBIT 5-4 SAMPLE COST/BENEFIT WORK SHEET

One-Time Costs	Cost	Year Cost Incurred	Life Expectancy	Salvage Value
Feasibility study, system design				
Hardware purchased				
Field equipment				
Communications equipment				
Cost to rework yard yard layout				
Site conversion/modification				
Power supply				
Air conditioning				
Training				
Implementation and installation				
Disruption of current operations				

Recurring Costs	Cost	Period
Maintenance/replacement		
Energy		
Space rental		
Communications		
Supplies		
Hardware lease payments		
Additional personnel		
System operator		
Maintenance/repair		
Programmers/analysts		
Management		

- Humping speed--Greater blocking capacity may lessen the need for additional hump yards in the future. Increased yard throughput means each car spends less time in the yard and is better utilized.
- Hump and crew utilization--Savings in personnel and energy may be realized.
- Track occupancy--Track utilization decreases the need for additional yards in the future and provides a better return on the investment in the current yard.
- Switching reliability--More time is available for humping when the humps need not be closed for removal of misswitched cars.
- Proper coupling--Fewer uncoupled cars mean that tracks need not be shoved as often. Fewer overspeed impacts lower costs of damage to lading and cars.

5.2.5 Select Option

The project team next compares options to decide on a particular one. A number of nonquantifiable factors such as management structure, data processing centralization, hardware compatibility, and union agreements usually become important. Once an option has been selected, the project team can provide management with a budget and schedule; and because the physical requirements have been defined, the team can begin the hardware study.

Corporate policy affects the relative weight given to hardware costs, crew efficiency, information access and service to shippers,

Table 5-1

JUSTIFICATION FOR PROCESS CONTROL COMPUTER USE

Improvements Over Conventional PC	How this Is Done by PC	Effective Overall System Performance
Few misclassifications	Car identification and switching logic are more efficiently done with computerized PC	Less time lost Increased throughput Reduced engine time
Automatic generation of misroute report	The PC computer keeps track of misroutings. Therefore, it can produce a report on request.	Eliminates time ordinarily required for searching for lost cars Labor savings
Higher humping rate	This is achieved through better hump engine speed control.	Increased throughput of the yard
Better track utilization	The PC computer maintains track occupancy and inventory information on a real-time basis, which provides a more dynamic track assignment scheme.	Increased storage capacity of the yard Increased throughput A more sophisticated sorting and blocking scheme may save time for the downsteam yards
More timely car location information	The PC computer maintains track occupancy and car inventory information on a real-time basis, which provides a more dynamic track assignment scheme.	Quicker response to customer inquiry Reduced search time in locating cars
More optimal retarder control	Automatic retarder control can provide a higher level of accuracy ($\pm$ 1/4 mph) and consistency in the speed at which a car leaves a retarder than can human retarder operators. In manual operations, improper switch list information may cause cars to be retarded inadequately or excessively. This results in car spacing problems and overspeed impacts. In manual operations, there may be higher average coupling speeds because of a tendency for operators to release nonfragile lading at higher speeds than fragile lading. Overspeed impacts weaken couplers and yokes; computerized control reduces these occurrences, thereby reducing the number of damage claims and car repairs. Car spacing problems are to a great extent alleviated by more accurate retardation, resulting in fewer stalled cars.	The reduction of underspeed (improper spacing) can reduce crew and engine use. Both improvements mean the saving of lost time due to corrective actions, which amounts to increased yard throughput.
Early detection of bad-order cars	Automatic detection of bad axles by wheel detectors will cause automatic detour of the bad-order cars to the maintenance area before classification.	Saves time in "digging" the bad-order cars out from the classification tracks later Saves crew and engine times Reduces damages later if the bad-order cars are not left undetected
Automatic integration of speed control	The PC computer can sense and utilize speed control data from the bowl area to adjust speed control parameters at the crest area or vice versa.	Increased throughput Saves crew time and engine time and energy
Quicker and more positive identification of inbound cars	The automatic processing of inbound train consists and display of switch lists dynamically as the train is being humped, plus the verification of individual car identifications by comparing wheel detector input with computer-stored records, give quick and accurate identification of cars.	Reduction of misclassifications, which means better throughput and fewer crew and engine requirements Less error in car inventory and car location data results in better management and customer relations.
More automatic area train protection and release	The automatic setting of switches to lock an area in order and turn it over to local control and the release of such control are done by the PC computer via CRT terminal interaction.	Saves manual switching time Reduces routing conflicts, thereby reducing crew and engine time, and increasing throughput Reduces car damages

Table 5-2

JUSTIFICATION FOR MANAGEMENT INFORMATION COMPUTER USE

Improvement Over Conventional Methods	How this Is Done by MIS computer	Effective Overall System Performance
Up-to-the-minute detailed inventory	Disk-based PICL maintains track occupancy/order and car inventory information.	Quicker response to customer inquiry Reduced car search time More accurate train makeup information, decreasing errors and reswitching Provides information for more accurate demurrage accounting
Improved data entry retrieval	Data entry/retrieval using formatted CRT screens.	More accurate data entry Easy, fast retrieval of information used in decision making Reduction/elimination of the expense of some preprinted forms
Improved data access and communications	Low-cost CRT terminals located throughout the yard.	Better inventory control when responsibility is given to respective department More accurate input as data source enters information Personnel time savings because of easily accessible data
Better recordkeeping	Disk-based data storage with automatic recordkeeping from event transactions. Tape backup for permanent record storage and automatic audit trail generation.	Reduced expense of preprinted forms Less voluminous record storage Increased data accuracy and accessibility Fewer clerical personnel
Better reporting	Computer processing power can produce reports that are unobtainable by manual methods. Information can be presented via CRT displays or as hard copy at multiple locations. A well-defined data structure can be easily manipulated for new report production.	More accurate and comprehensive reports Reduced manpower for report production Reports are seen earlier by decision makers
Automatic data transfer to systemwide centralized computer	Computer-to-computer communication automatically transmits inventory and other information.	Reduced manpower needs More accurate data transfer
Computer-assisted waybill generation	Computer-generated waybill format on CRT automatically enters repetitive information. There is on-line rate calculation by the computer and the completed bill is automatically printed.	Fewer waybill errors Reduction in clerical effort Faster processing speeds for revenue collection
Automatic information interchange	Waybill, consist, and interchange reports and information can be transmitted to another line's computer automatically.	More accurate information Reduction in clerical effort Faster processing speeds for revenue collection
Automatic switch list generation and dynamic track assignment	Automatic switch list generation from inventory records of car destination and track/block assignment. On-line computer assistance for track assignment.	Reduced personnel effort Quick transfer of information to locations in the yards More accurate switch lists to reduce car-handling errors Increased throughput A more sophisticated sorting and blocking scheme may save time for downstream yards Fewer throat conflicts, better trim engine and crew usage
Crew assignment and tracking	Work requirements and predictions are used to assign work tasks to the proper employees and crews. Accurate records are kept and the computer monitors work rule violations.	Efficient work force utilization Less overtime pay Reduction in penalty pay costs Accurate and detailed worker productivity reports Automatic transfer of work records to payroll at the central computer
Outbound train makeup	Current and predicted inventory information is used to determine optimal train makeup, engine use, and trim operations.	Optimal trim engine and crew usage Decreased congestion in yard throat area Efficient use of line haul engines and crews

service for other railroads, information timeliness, and the different methods used to measure the operational efficiency of a yard. The centralization or decentralization of management usually parallels the structure of the authority over and location of computer resources. Railroads that tend to have a high level of local control usually offer easier information access, have software autonomy, and place a greater emphasis on automatic yard decision aids.

Management within the yard and the autonomy of individuals and departments influence the access to computer resources at the local level. Within a yard, the flow of information can be centralized through an EDP department and computer room, restricted to "privileged" users by requiring security and an access code to CRT terminals and files, or it can be decentralized by allowing individuals to enter information to add to or change the data base directly via CRT terminals located throughout the yard.

Union agreements at different railroads and yards may also influence system design. Computer operators, required for a large centralized system, may be classified as clerks. If a decentralized replacement is suggested, in one interpretation clerks may be required to enter or change data from any CRT terminal in the yard. For example, car control actions made by operations personnel on a PC computer may also change inventory records on the MIS computer. In some railroads, DP personnel must be classified as management personnel. Therefore, the installation of a new or larger computer system may require more "managers" to work in a yard than are allowed by labor agreements. A restriction on crew size may remove any labor savings attainable from the installation of a PC system.

One of the main institutional considerations in yard computer applications is the attitude toward computer failure and downtime. In certain railroads, the management philosophy is that a yard must operate at maximum capacity all the time, so the railroad is willing to pay for a highly redundant system that guarantees no loss of computer capability. Other railroads could tolerate a degraded mode of operation for a short time, thus requiring a less costly backup computer system capability. Attitudes toward downtime lead to important considerations of product support and maintenance. Vendors with more extensive support services covering a large geographic area may have a distinct marketing advantage. Thus, management's perceptions of the relative importance of individual yard functions are important in the analysis of trade-offs between complete and partial redundancy of the hardware that automates these functions.

Another institutional consideration is whether the field DP service personnel are to be located at headquarters, in regional facilities, or at the yard. Having field DP personnel at the yard facilitates a decentralized system architecture; having them at headquarters facilitates a more centralized system architecture. Labor agreements concerning yard personnel influence whether field DP personnel are located at yards or centrally.

Currently, the management philosophy of most railroads tends to be "risk adverse." The railroads tend to choose computer hardware with less capability but for which a vendor can guarantee computer support and maintenance. However, if the computer skills and technical know-how are available within the railroad, management can adopt more innovative hardware and applications.

Management's perception and understanding of the future technological trends and cost-performance trade-offs in computing capability, computer memory, telecommunications, and software greatly influence the system architecture. Those managers who believe the principal cost will be in software arrive at decisions differently from those who believe the greatest costs will be in telecommunications or in computing capability. For example, a railroad may choose centralization of computer operations because of excess capacity in common carrier communications lines along its right-of-way.

5.2.6 Functional Specifications

As part of system design, the project team develops in the analysis phase a set of functional specifications. In addition to the use in systems design, specifications advise the users how the new system will affect them. Functional specifications include:

(1) A statement of preliminary equipment requirements.

(2) A list, as well as samples, of all reports and other output from the system, documentation of the flow of paperwork, frequency of reports, number of copies required, distribution of these copies, and any special requirements.

(3) Performance requirements, which serve as source material for subsequent test specifications and acceptance tests.

(4) Information on each file or data base and its probable contents.

(5) Processing requirements that discuss topics or functions to be performed by the users and the computer.

(6) Proposed organizational changes and changes in clerical procedures. (Staff requirements for the next few years may be projected at this time.) If interactive systems are to be used, scripts are developed to specify procedures for terminal users.

Functional specifications are revised until the project team and user representatives agree. This level of detail is sufficient for the procurement of stand-alone PC systems from a turnkey vendor.

The specification will be assembled from documents and exhibits compiled from each step of the analysis, data dictionaries containing definitions of system interfaces and files, and a complete description of the internal logic of each process of the logical flow. The specification will also document the partitioning of functions among computer systems and subsystems and individuals and departments. The specification will also include performance targets for systems installation and acceptance.

5.3 SOFTWARE DESIGN PHASE

In the software design phase, functional specifications are expanded into a system specification to provide more details on man-machine interfaces, user procedures, performance requirements, the structure of software partitioning and modules, and more detailed design of data files. Hardware requirements are important inputs from the hardware study. If a PC system is to be purchased, the software design phase may not be required before the procurement process begins. In some cases, the software design phase is completed by the vendor.

5.3.1 MIS Software Characteristics

5.3.1.1 Operating Systems. Many of the advances in and distinctions between yard management computer systems have been in the development of operating systems. Older computers use a serial-batch operating system limited to execution of a single job. This is adequate for simpler applications, such as the card-PICL system.

Newer yard computers must concurrently perform multiple processing tasks, provide on-line data input and retrieval, and service multiple communications lines. A multiprogramming or time-sharing operating system can provide these services (Section 2.4.1.3). A yard management computer may operate by following a predetermined cycle of tasks, or it may operate in response to random interrupts requesting processing. If the inventory program receives information directly from the yard PC computer, its operating system should be interrupt driven to ensure prompt servicing of the PC computer, which may have limited storage capacity.

When multiple programs are run concurrently in the same computer, a priority structure is needed to ensure the speedy processing of important or time-critical jobs. For example, an on-line application, such as car location inquiry, should take priority over switch list generation, which may in turn take precedence over production of periodic management reports.

Foreground/background processing is one technique for efficiently processing high- and low-priority tasks together. High-priority on-line jobs are processed in the "foreground"; that is, they are serviced interactively and are given a high percentage of CPU resources to ensure quick processing response. Low-priority batch jobs can be processed in the "background": Because output is not needed immediately or interactively, the percentage of CPU allocated to background jobs depends on the needs of the foreground jobs.

To provide the equivalent of a larger main memory to permit running larger programs on a smaller computer, virtual memory or program overlay techniques may be used. Overlays allow use of the same memory location for different programs, and the virtual memory system uses disk memory as an extension of the main memory.

If the computer configuration provides redundancy in case of failure, the operating system must be designed to detect the error, switch hardware, and restart applications. Failure recovery may be designed to be completely automatic or may require various degrees of manual intervention.

5.3.1.2 Information Storage and Access. Yard MIS applications require large amounts of data to be stored. An inventory program relies almost exclusively on file access and manipulation routines. Files must be designed to use space efficiently while remaining easily accessible for frequent inquiries and changes (also see Section 3.2.2). An efficient inventory file structure should:

- Minimize storage needs.
- Minimize the search for storage locations when adding records.
- Easily monitor car locations.
- Easily provide car information by track-standing order.
- Minimize the search effort to locate a particular car.

Missouri Pacific's YATS system uses three different files to store car ID, car records, and car position. The car index file (CIF) stores car IDs in an easily accessible order. A pointer for each car indicates the location of the car information in the car record file (CRF). A third track inventory file (TIF) stores individual pointers to the car record file in the standing-track order of the yard. This file structure allows the bulk of the information to be stored randomly and efficiently in the car record file while remaining accessible by ID or location through the CIF and TIF files.

Union Pacific uses a similar file structure. The difference is that the TIF includes both a pointer and the car ID, so that a car can be moved in inventory by specifying the track and car ID as opposed to the "track sequence number." The software for yard inventory systems is very similar to standard inventory packages. Information can be stored and large files can be accessed efficiently using data base management software.

Journal files are necessary for an accurate record of yard operations. Files can be copied periodically to act as a checkpoint, or a record of transactions can be made to act as an audit trail.

5.3.1.3 Input/Output Terminals. If the project team decides to provide easily accessible CRT or teletype terminals throughout a yard, display software must be obtained. Efficient processing, good communications, and an interactive multiprogramming operating system enable many users to access a computer simultaneously. Once hardware requirements are met, display software and data base accessing procedures must be designed. Terminal software should:

- Provide fast data input.
- Promote accurate data input.
- Be nonthreatening and easily acceptable to yard personnel.
- Present displays in an easily readable form.

The first consideration in the design of user procedures is the method to use for data input and output. One method is to designate an English type of command, such as "Enter car ID." If many commands must be memorized, however, problems may develop. A popular solution has been to call for a "menu" of commands. The user selects a procedure from the menu by specifying a number or symbol. In Southern Pacific's West Colton Yard, the user may ask for a menu or enter the required function symbol directly. In the selection of some functions, the user receives a list of applicable choices. An example is a request for an inbound train consist. This request prompts display of a list of the identities of all incoming trains, and the user is requested to select a single train by placing an "X" beside the ID number.

If processing power is sufficient, large, complicated input procedures can be presented on a CRT display by a self-tabulating input form that requires the user only to "fill in the blanks." The user is queried for information, and after each field is entered the computer can immediately check for data entry errors and demand immediate reentry if an entry is incorrect.

Report generators are software tools the user can use to design data output formats. Report generation programs interactively query the user on data fields, sorting, and presentation. This enables the user to quickly design a "one-time" or permanent management report.

5.3.1.4 Communications. If the MIS computer must communicate with another computer, software may be needed for the interface between the two machines. Messages must be reformatted, incoming messages must be decoded, and messages must be queried to ensure proper timing.

Computer-to-computer communication in classification yards is required for internal-yard-distributed processing, MIS-to-PC computer linkage, and sending and receiving external messages. Distributed processors must communicate large volumes of information quickly. The linkage between machines may be a high-speed communication line, a shared communications bus, a direct-memory access channel, or a shared memory.

The PC computer, as a real-time system often with little memory capacity, must be able to communicate immediately to the MIS system. The MIS computer must therefore buffer the stream of incoming messages from the PC system until they can be processed. When the PC and MIS computers are separated by a great distance, messages must also be formatted for transmission.

The communication to other yards and to the railway central computer need not be as time critical. Messages may be stored and sent periodically as a batch to an outside location. If a direct link between railways is not used, messages can be routed through a common computer such as the Association of American Railroads TRAIN II communication network.

5.3.1.5 Software Languages. The software language chosen for use in the yard MIS computer system should be well suited to file accessing, file manipulation, and report and display generation. For management applications, considerable scientific processing and mathematical computation are not required.

In general, MIS applications on a large computer fit well with high-level business- and file-oriented languages, such as COBOL. Programs written in a high-order language are preferred because they are easier to develop and modify. Programs written in a standard machine-independent language are transferable to other machines and can take advantage of software developed on other machines. If a microprocessor is used for an application such as communications processing, a lower-level language may be preferable. Because of the smaller memory and processing power, microprocessors benefit from the efficiency of assembly languages.

5.3.2 PC Software Characteristics

PC applications have traditionally been written in low-level assembly languages. Assembly languages have been preferred over higher-level languages such as FORTRAN because of their faster execution times, compact code, and ease of direct access to I/O data. For a higher level language to be successfully used, the compiler program must efficiently compile programs into quickly executable code and permit detailed manipulations for concentration and verification of the incoming and outgoing field sensor signals.

Because few PC applications require specially designed hardware, general-purpose computers are adequate for railroad yard applications if they are equipped with an efficient real-time operating system. One capability that is valuable for railroad PC applications is priority-structured interrupt-driven task scheduling. Software development systems usually require a large disk-based operating system. If the software is developed on another system, a smaller, simpler memory-based real-time operating system is usually sufficient. If a high-order language is used for programming, a valuable feature is to allow lower level system and I/O commands to be called from within application programs.

5.3.3 Software and File Redundancy

When MIS and PC functions are divided between two computers, the PC system relies on a number of MIS functions. In normal operation, the MIS is expected to periodically provide new switch lists assembled from the receiving yard inventory and classification table. Continuous humping therefore depends on the MIS computer's timely completion of this function. This dependence can be avoided by having redundant functions and files in the PC computer.

Two schemes may be used to provide this redundancy. The first is to store a number of upcoming switch lists in the PC computer, and the second is to store the receiving yard inventory and the classification table in the PC computer. The choice depends on the expected repair time for MIS computer failures and available storage in the PC computer. If the repair time (MTTR) is short or the MIS can be quickly reconfigured to a downgraded single computer with switch list generation, then only a small number of switch lists need be kept to ensure continuous humping. If longer or complete failures of the MIS computers can be expected, more software redundancy is required. In this case, the receiving yard inventory and a classification table are stored in the PC computer and are used by redundant software to generate switch lists when required. In both cases, the final inventory location of humped cars must be stored on disk or as punched cards until the yard inventory can be updated on the MIS computer.

5.3.4 Software Specifications

System specifications for software include the basic design for the programming system and subsystems. Before programmers begin working, they must understand the system objectives and specific responsibilities that they will have in developing the software. To provide this information, the project team should prepare a written narrative that describes the new system. This narrative should ideally contain a description of the type and extent of the system, the general design concept, and a brief statement of the purpose of each program. This narrative is used for actual program development when software is completed in-house. If software is to be supplied by an outside vendor, the vendor may complete the narrative. It then can be used to again confirm user acceptance and understanding of the proposed system.

After the system narrative has been completed, systems analysts write specifications for individual programs that state what the programmers must accomplish. First, the functional specifications are broken into their component parts so that system modules may be defined. The logical control and sequences to execute each function are then identified. Detailed design should specify data names, formulas or specific logic for complex routines, and record layouts. All conditions should be covered, so flowcharts, data flow diagrams, and/or decision tables may be helpful at this point. System specifications should be finalized and approved before programming begins. Often a committee of people from different departments and yards that have a stake in the system is formed to review the design.

The project team should prepare a procedure for modifying the final specifications; this procedure should include reviews by the affected users as well as management. The procedure can be formalized through request-for-change forms that document the requested change, the person and department requesting the change, the associated benefits and costs, and whether the change should be effected immediately or can be postponed. A formal review and modification request procedure is required for software developed by a vendor but should be completed by an agreed-upon stage of development.

5.4 HARDWARE STUDY

The goal of the hardware study is to design and specify computer hardware and other equipment. An MIS hardware study covers new hardware selection, design of additions and upgrades to the present system, and an evaluation of the effect on and interface with the existing system. The hardware study for the PC system is not likely to be as comprehensive because in most cases PC computer hardware is part of a complete turnkey package. In the analysis phase, the project team may have already identified the design, constraints, interface, reliability, and performance information required. A PC hardware study is likely to concentrate on interfaces with the existing or projected MIS system, reliability requirements, and redundancy solutions to be implemented by the selected vendor.

The hardware study is begun during the analysis phase to provide general hardware trade-offs and costs for use in comparing alternatives (Step 6, Figure 5-4). Hardware specifications are completed in detail after the analysis phase.

To begin the hardware study, the analyst reviews the data on the current volume of classification operations and information processing that were gathered during the first step of the analysis phase, description of current operations. By comparing the functional diagrams of the current system and the selected alternatives, the analyst identifies the changes between systems, added functions, and newly computerized functions. Any expected increases or decreases in the future volume for any function must also be noted for determining the size of various hardware components and evaluating the need for software. The hardware and software for any function that is expected to expand in the future should be sufficient to incorporate the added volume, unless it is cost-effective to add additional capacity later. The results of the analysis phase are also used to examine the structure of the software, including file size and structure.

The common bottlenecks encountered in designing a computer system are in the processor, in the disk drives, and in communication to the user in the yard, other yard computers, and to computers outside the yard, such as a railroad systemwide computer.

5.4.1 Hardware Requirement Analysis

To estimate the required size of computer hardware for any configuration, mathematical modeling analysis, simulation, or empirical formulas can be used. Explaining these methods in detail is beyond the scope of this manual. More detailed explanations and additional references are available from the following sources: J. Martin, "Design of Real-Time Computer Systems" (1967); the articles in Computer (1980) and Computing Surveys (1978); E. Yourdon, Design of On-Line Computer Systems (1972); and Hartman et al., Management Information Systems Handbook (1968).

With mathematical modeling, assumptions are made about the method the operating system will use to sequence and service programs. The simplest is to service all programs equally on a first-come/first-served basis. Other schemes are to sequentially process jobs, process jobs with higher priority first, handle I/O requests first and use the remaining time for processing, or allot each job a predetermined time for processing before switching to another job.

The servicing scheme is represented by a mathematical model, as is the arrival rate of data and processing requests. Figure 5-20 represents a simple model of processor servicing. A Poisson distribution is often used to represent a random distribution of input from user terminals. Because of the random nature of request arrivals, queuing theories are used to estimate the waiting time associated with high system loads. This will provide an estimate of the percentage of response times that will be at an unacceptable level for any given processing capability.

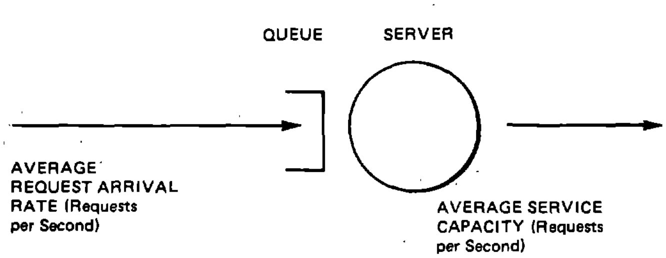


FIGURE 5-20. SAMPLE QUEUING MODEL OF PROCESSOR SERVICING

The first model constructed is of the complete configuration and includes a representation of the interaction between significant resources; Figure 5-21 is an example. Information is gathered on device characteristics, the structure of expected software, and operating system and CPU scheduling methods. Data are also gathered on the present and expected work loads including volume of user input and requests, data storage resource requirements, individual device utilization, and the required system response time. This information is available from the operational data gathered (Section 5.2.1.1).

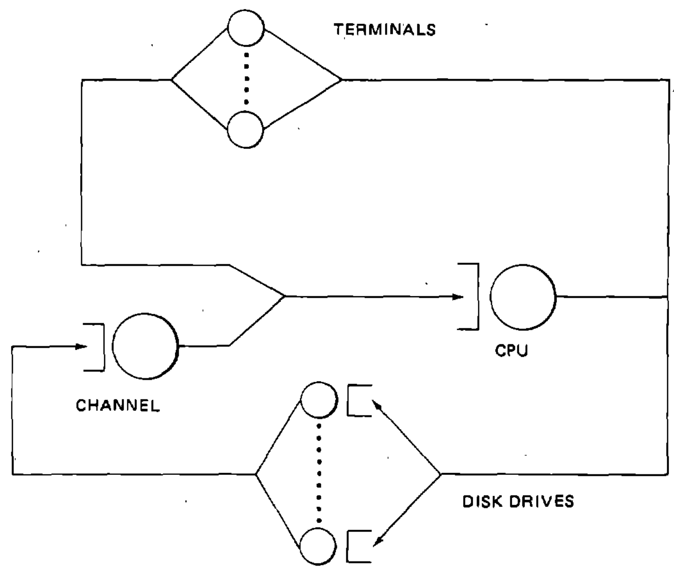


FIGURE 5-21. SAMPLE CONFIGURATION MODEL

For any particular configuration, the following hardware characteristics must be identified: instruction timing, main memory size, main memory cycle time, cache size, cache cycle time, memory access time, number of disk drives, minimum and average seek time, average rotational delay, disk transfer time and volume, priority servicing for hardware and software requests, the number of terminals, and the scheme for terminal and other I/O servicing. The analyst must also estimate disk accesses and processing time per transaction. A number of analytical models (BEST/1, CADS) are commercially available.

In contrast to mathematical models, a simulation model uses a computer to estimate characteristics of hardware in the expected operating environment. The program uses input and processing characteristics to generate such statistics as maximum and average queue sizes, distribution of user response times, hardware utilization, storage requirements for buffers and queues, and maximum throughput (Martin, 1967).

Some hardware vendors offer simulation packages for their specific hardware--for example, SCAPE for IBM systems using MVS operating systems

(Schiller, 1980) and the IBM VM/370 Predictor (Bard, 1978). Other independent packages are available commercially, such as SCERT (Hermann, 1967). In addition, a number of simulation languages are commonly used--SIMSCRIPT, SIMULA, SIMPAC, and GPSS III. Simulation models can also be written using standard languages such as COBOL and FORTRAN.

The use of common empirical formulas is a third hardware requirement analysis method. A number of common factors have been observed with on-line business computer systems, and some may be applicable to yard computer systems. For example, the average typing rate at a terminal is one or two characters per second. It has also been observed that a transaction is completed every 30 to 60 seconds. These numbers, used in conjunction with the work load estimate of clerical functions in a yard, can roughly indicate character and transaction arrival rates.

Another observed reationship is that a minimum of 8 to 10 or an average of 15 to 20 disk accesses are required for processing each transaction (Yourdon, 1973). This relationship varies with different programs; some complex reporting programs may require many more disk accesses on the average because of the file access methods, such as indirect addressing used to access multiple files.

Another empirical formula often used is $n = T/P + 1$, where T is the arrival rate of transactions, P is the average processing time, and n is the obtained estimate of the number of active terminals a system can support (Scherr, 1967).

5.4.2 MIS Hardware Characteristics

The hardware characteristics of a particular classification yard MIS are determined by the applications that are to be processed. This section discusses how each type of hardware component is used and how it is affected by each application.

5.4.2.1 Central Processing Unit. The CPU must be chosen so that it is capable, within time constraints, of fully processing the work load. The PC computer must be capable of operating as a real-time system, whereas the MIS computer need not be time critical in all its applications. All the MIS applications may be completed using a batch operating system as long as no on-line inquiries must be returned to either a terminal or the PC computer. If the MIS computer is a batch system, the CPU need only be capable of processing the average load of a 24-hour period. Batch processing is adequate for daily inventory records and management summaries.

Real-time applications require that peak loads be processed quickly enough to return time-valuable information. If the PC computer relies on the MIS computer for such information as switch lists, this information must be returned promptly or yard throughput will decline. If CRT terminals are to be used for on-line inquiry and data entry (such as inventory inquiries, inventory movement, yard status reports, on-line dynamic track assignment, and the like), the processing capacity must be great enough to provide adequate response times. In addition, the CPU must be capable of processing a large amount of character and file manipulations for inventory applications. The CPU must quickly process a large number of file transfers for inventory information transfers between PC and MIS computers and interyard information transfers require.

Decision and planning aids require a powerful CPU. Scheduling/utilization algorithms, statistical analyses, and yard operation simulations may also require considerable CPU power. If these functions can be completed off-line during non-peak hours, total CPU power requirements can be reduced.

Word size becomes important when large amounts of data must be transferred quickly and stored efficiently. Word size is not as important for arithmetic processing because there are few floating point calculations for inventory, management reports, and interyard information applications.

CPU sizes vary with application and yard size. Southern Pacific's Terminal Control Computer services several yards with real-time inventory, management reporting, and other functions using large mainframe computers. The Yard and Terminal System (YATS) of the Missouri Pacific Railroad (MOPAC) was designed for a yard handling 500 to 6,000 cars per day. It supplies a track-standing inventory, management information report, task tracking, and an on-line data base for operations analysis using a 16-bit minicomputer with a main memory of only 80K to 128K words (160 to 256 KB). Southern Railway's Terminal Information Processing System (TIPS) consists of two 80K minicomputers for inventory and switching control, waybill generation, industry inventory, demurrage accounting, and work-order tracking and analysis (Whitehead, 1977).

5.4.2.2 Mass Memory. Mass memory normally consists of magnetic disk memories or, in smaller yards, flexible diskette memories. By far the greatest use of mass memory in yard MISs is on-line inventory files. The size of mass memory required varies directly with the maximum number of cars in the yard inventory, the amount of information stored for each car, and the length of time the information is on file. Examples of disk size requirements are Southern's TIPS, with 40 million bytes (MB), and MOPAC's YATS with 8 MB to 80 MB for 500 to 6,000 cars and its INCOTERM System with a 1 MB diskette for 300 to 500 cars. Because of lower hardware costs and greater reliability of the Winchester type (sealed) of disk drives, MOPAC has been replacing the diskettes with 24 MB drives.

Inventory records can also be kept on cards or tape, but this limits the speed of processing and memory access. Tapes are used extensively for archival record storage.

5.4.2.3 Card and Printer I/O. In the first yard computer systems, cards were used almost exclusively for data input, mass storage, and data transfer. Printers provided the data output and a permanent storage medium. An example is the Perlman Yard of the New York Central System, which was built in 1968 (Karvwatt, 1968). This yard was equipped with card readers, card punches, and high-speed printers to enter and retrieve bulk data from the computer. The primary input were train consist card decks and the primary outputs were switch lists, corrected switch lists, and track inventory cards. All yards using card-PICL systems rely on card and printer devices.

In many recently built yards, data input, mass storage, and data transfer functions have been taken over by CRT terminals, disk and tapes, and data transmission carriers. Printers are still the best and fastest method to receive a hard copy of information from the computer but CRT keyboards have replaced cards and printers for data input and inquiry. In Southern Pacific's West Colton Yard are 32 CRT terminals, compared with only 11 printers and 9 teletypes. The printers are used for printing large-volume reports, and the teletypes are used for logging alarm and maintenance messages. A card reader/punch acts as a backup for the communications link between the crest control (PC) and terminal control (MIS) computers. In Santa Fe's Barstow Yard, high-speed card punch/processor/printers are backups for the yard computer to transmit data to the hump computer and to Santa Fe's central computer in Topeka (Progressive Railroading, 1976).

5.4.2.4 Data Input and Inquiry Terminals. As computers have become more powerful and operating systems and software have become more sophisticated, price performance has greatly increased. Thus, classification yards have been able to afford real-time and on-line MIS. With installation of on-line systems in newer yards, most card punch equipment has been replaced with CRT terminals and other data/inquiry terminals. CRT terminals provide a fast, accurate data entry system that mirrors back input for immediate self-checking. Low-cost CRT terminals can be strategically located throughout the yard, allowing easy access to speedy data input and inquiry. With entry of data into a yard data base, much of the intrayard communication can be completed using only CRT terminals, saving time and eliminating much of the transfer of written data. When needed, smaller terminals can be moved to any place where telephone lines give computer access.

In card entry systems, a difficult, rigid, column-by-column data entry format is required, but data entry via CRT terminals can be formatted to provide easy man-machine communications. Data entry programs can query the user to select a data entry type from a set menu and then prompt the user for each specific item of data and then check the entered data for formatting and logic errors. This type of advanced data entry formatting reduces errors and training time.

Terminals may be displays only, or they may include a processor that can relieve the main computer of some functions--data entry formatting, for example. If the terminal also has internal storage, data can be condensed and stored for later, more efficient, less costly transmission. The main computer need only occasionally query the terminal for data. Intelligent data entry terminals are cost-efficient because they save high communications costs or off-load tasks to postpone or reduce an upgrade of the central facility.

CRT terminals are used for on-line inventory inquiry and management reporting and as on-line decision aids. Special displays can be used for the graphical representation of yard inventory distribution or a real-time representation of the humping sequence.

The distribution of CRT terminals differs from yard to yard. Southern's Savannah Yard has seven CRT terminals and two printers located in the yard and agency office, one CRT terminal and one printer in the yardmaster's tower, and three printers in the switching area (Whitehead, 1977). In Union Pacific's Terminal Information System, CRT terminals and printers are distributed among the yard office, the yardmaster's office, the mechanical facility, the TOFC ramp, and terminal control (Anderson, 1977).

5.4.2.5 Long-Distance and High-Volume Data Links. Data links to remote locations are used in three situations. The first is where the yard MIS computer is located in the yard but must communicate with the railroad's central computer. Examples are Santa Fe's Barstow Yard microwave link to Topeka, Louisville and Nashville's Strawberry Yard data link to Jacksonville, and Southern's Sheffield Yard 2,400-bps data link to Atlanta.

Alternatively, the yard management computer itself may be located at a remote central site. Southern Pacific's Terminal Control Computer (TCC) is located in San Francisco--more than 400 miles from its West Colton Yard. Direct access is provided for the crest control computer and 52 printers and CRT terminals by a direct microwave link. These devices operate as if the computer were located physically in the yard.

The third area of data communications is interyard and interrailroad communications. Rail lines communicate via direct computer-to-computer links, using teletypes or high-speed printers, or through an intermediate computer such as the Association of American Railroads TRAIN II communications network. An example is paperless interchange reports (Section 3.1.10.2).

The high volume of data on waybills and consists requires high-volume I/O channels for each computer system. One of the greatest difficulties of high-speed communications is the design of a compatible interface between computers. Data formats, protocols, and speed must be matched for direct communications. In most cases, standard hardware can be purchased and minor programming changes made to provide a satisfactory interface. In more complicated situations, an intermediate communication processor is used.

5.4.2.6 Microprocessors. Recent advances in computer technology have provided in smaller mini- and micro-computers that offer large amounts of processing power for relatively little cost. This less expensive computer power provides the opportunity to use independent processors to process parts of current applications as well as for previously untried applications. A number of applications now using minicomputers are applicable to microprocessors.

Microcomputers can be used in two ways. One is to group them as dependent processors to share the equivalent work load of a single MIS computer. By distributing the processing among many computers, the failure of one component has a lesser effect on total system operations. In a system of General Railway Signal Co. (GRS), five minicomputers are used for retarder control, automatic switching, process control backup, MIS applications, and MIS backup (DiPaola, 1973).

Microcomputers can also be used as independent or semi-independent processors completing specific MIS tasks in the yard. On-line decision and planning aids are especially adaptable to microcomputers. With use of an independent computer, speedy response times can be maintained without affecting total MIS throughput. Decision aids are also sufficiently independent of other applications to permit their development and execution on a separate computer system. An example is Union Railroad's Yard Crew Management System, which uses two independent minicomputers.

Inexpensive microcomputers allow not only the development of new applications, but also the use of computers in new locations. Installation of computers in smaller yards and agent offices will soon become economically possible, with the use of smaller independent MIS systems or of intelligent data entry terminals capable of communications preprocessing, data base inquiry, and output formatting. An example is the MOPAC's development of a remote job entry (RJE) system for small yards (one or two tracks) and agents (Lynch, 1977).

5.4.3 PC Hardware Characteristics

In most cases, a PC system is purchased in a hardware, software, and installation package from a PC systems vendor. Nevertheless, the goal of this manual is to familiarize railroad computer system users with the design of PC systems so that they will be more knowledgeable in systems design and selection. Thus, this section discusses important PC hardware characteristics, including external I/O interfaces, CPU speed in processing real-time transactions, communications links to other computers, and a high degree of operational reliability. Applications programs are the most important PC software. They are developed from the user-defined functional system specifications and are discussed in Chapter 4 and Section 5.3.2.

During the earlier stages of computer development, the PC computers were considered as a special class and were designed differently than general-purpose computers. Examples of these special-purpose computers are the IBM 1130 and the Honeywell 316, 416, and 516. The unique features of these computers are their word size, special front-end I/O interfaces, dynamic interrupt features, and real-time-oriented computer instructions.

As computer technology became more sophisticated, the difference between the special-purpose computers and the general-purpose computers gradually narrowed. Today, except for a few applications (e.g., space flight, military command-and-control), the PC computer is basically a general-purpose computer, with the addition of certain readily available operational features. This section presents an overview of these features as they apply to railroad classification yard operations.

5.4.3.1 I/O Interfaces with External Sensing and Control Devices. The PC computer requires certain external I/O interfaces for accepting real-time data from wayside equipment, for example, switch detectors, weigh scales, and radar units. The PC computer is also required to send out instructions to the control equipment for switches, retarders, and the hump.

The output from field sensors is usually in analog form; for example, the passing of an axle over a set of wheel detectors in a certain direction causes the sensor unit to emit a level of DC voltage (VDC) within a predesignated range (for the other direction, the voltage output would be within a different range). The absence of a car produces a zero VDC. Typically, these analog signals are first channelled into a multiplexer and then sent through an A-to-D converter. The digitized data are then read into the PC computer for further processing.

The flow of control signals from the computer to the field equipment is essentially the reverse, that is, the digital information from the computer goes first through the D-to-A converter, then through the multiplexer, and finally to the line drivers (or data distributors) for sending to the various wayside control devices.

5.4.3.2 Real-Time Transactions. In railroad yard PC applications, real-time control implies that the PC computer can complete all activities; it samples field surveillance data, digests these data, makes control decisions, and sends out control signals in time to influence the next cycle of field operations relevant in maintaining a predetermined set of operational objectives. For example, in an automatic switching process, a cycle may be defined as the elapsed time between the passing of consecutive cuts at a fixed wheel sensing station. In satisfying this objective, the PC computer requires a high-speed CPU, a large internal memory, and multichanneled I/O devices.

Another important requirement is a powerful computer instruction set. It should offer several levels of priority interrupts, including the automatic handling of inputs from real-time clocks. Instructions should be included to allow bit position manipulation and logical operations. An additional requirement is that the PC computer offer a sufficient number of parallel registers to permit transfer of the maximum amount of field data within one cycle time. Computers with cache memories are very advantageous to PC operators because they permit rapid access to contiguous arrays, data files, and programs several times faster than the regular memory. This feature can easily apply to a computer-controlled humping operation, when all the car records belonging to one train may be kept in the cache memory for quick reference and updating purposes.

5.4.3.3 Communications Data Links. A yard PC computer not only is required to communicate with field control and surveillance equipment, but it also should be able to communicate on-line with other computers (e.g., yard MIS computer or computers remotely located in other yards or at the railroad headquarters). It also must communicate on a real-time basis with yard area supervisor, hump control operators, track inventory clerks, and computer operators. Therefore, the PC computer must use a variety of communication interface devices. Such devices include modems, line conditioning and equalization equipment, multiplexers and concentrators, switching processors, and communication processors.

The purpose of the modem is to modulate and demodulate a voice-frequency carrier signal with digital information. The most commonly used modems range from 300 bps to 1,200, 2,400, and 4,800 bps. The objective of line conditioning and equalization is to adjust the amplitude attenuation and envelope delay to within certain ranges for a more reliable transmission over voice-grade data communication lines. Multiplexing and concentration equipment is required for dividing a communication link into slices (or slots), each capable of carrying information from a separate source. The slices may be separated from each other by either time-division multiplexing (TDM) or frequency-division multiplexing (FDM). A switching processor is used to switch messages between various types of communications lines and handle various types of terminals, line speeds, codes, and communication protocols. A communication processor is generally more powerful than a switch processor in that it can also handle character assembly, network polling and handshaking, message queuing, error detection, error recovery, and code conversion.

5.4.3.4 Programmable Controllers. Recent advances in microprocessor technology have spurred the development of specialized microprocessor-based control systems. Programmable controllers are microcomputers specifically designed for process control applications that require diverse I/O communication requirements and reliability in a hostile environment. Programming is made simpler by mocking relay symbology and/or Boolean logic; a programmable controller can be programmed quickly to replace an existing relay-based control system (Coughlin, 1981). Programmable controllers are also designed for ease of maintenance (error detection and modularity) and for easy interfacing with computers for distributed control (Martin, 1981). Programmable controllers may be suitable for a number of applications, including process control at remote or satellite yards and the direct replacement of failing or aged relay-based PC systems in smaller yards.

A number of manufacturers offer programmable controllers. In some models, two microprocessors are used in the CPU. One efficiently completes mathematical and proportional/integral/differential (PID) functions, while the other acts as the control executive or sequencer. In another design, dual processors are used for parallel address selection, memory access, and ALU (arithmetic logic unit) operations. Sixteen-bit microprocessors are used in some models both for the CPU and as independent communications and I/O controllers. Modular I/O interfaces give added flexibility to communicate to a wide variety of devices.

The executive program of the programmable controller establishes the sequence of routines and I/O inquiries. Each I/O device or channel is serviced only once for execution of the sequence cycle. This requires that the programmable controller be fast enough to cycle one sequence within the critical time value for any controlled device. For example, if a programmable controller were used to control a retarder unit, the controller must cycle through the executive program fast enough to accurately measure the entrance velocity and control the retarder within the required time.

Programmable controllers are also adaptable to the control of processes at remote yards. Communications capabilities allow a programmable controller to act as a slave to a PC minicomputer at another site and communicate through telephone or other communication lines. Although programmable controllers are available with multiple processors and more than 48K bytes of memory, the lack of real-time interrupts and flexible programming capabilities may limit the future use to smaller custom process control applications distributed within the yard or at remote locations until lower cost, standard software as required by railroads is developed. In some cases, the use of programmable controllers as replacements for existing relay control will eliminate the need for a more sophisticated and costly process control computer.

5.4.4 Reliability

Reliability is an important consideration for configuration design. For each alternative based on a functional allocation, a number of redundancy methods can be used. As the yard operations rely more and more on the functions of the yard computers, their continued operation becomes more critical. Perhaps one of the most striking differences between a yard PC computer system and a yard MIS computer system is in their reliability criteria. A failure in an MIS computer (whether software or hardware) can generally be recovered later, provided that the computer is not used continuously throughout a 24-hour day and 7 days a week. Except for on-line data communication with other computers or terminals, a failure or shutdown of an MIS computer does not render immediate damage to other yard operations.

In contrast, the PC computer seldom works alone. Its functions are usually linked with other yard operations. Intermittent errors or a complete failure can cause immediate damage to the yard operation. For example, a faulty switching or speed control setting may cause derailment. A PC computer failure during an automatic humping operation may cause the shutdown of the entire hump, thus delaying the whole yard operation. The loss of a real-time inventory record due to computer failure would also seriously delay the switching and speed control processes because an accurate and up-to-the-second track inventory is needed for these operations.

Before software and hardware configurations are designed and selected, a set of reliability criteria should be developed and hardware/software failure detection and failure-mode operation specified.

5.4.4.1 Hardware Redundancy. A number of methods can be used for increasing the availability of yard computers, as was described in Section 2.3.9. One is to provide increased availability with a backup computer, a secondary system that can quickly take over at system failure. A backup computer with full redundancy is called a fail-safe system. If the second processor does not have equivalent capacity, it is called fail-soft; in this case, when the primary computer fails, the least critical functions must be shed for a degraded operation. These methods can be further refined by a choice of automatic or semiautomatic switchover.

The system at West Colton Yard has a hot standby configuration. Digital inputs from field sensors and controls are transmitted to both the primary and standby crest control computers. The standby receives and retains the car-tracking information as humped records until the primary unit has successfully transferred the information to the Terminal Control Computer in San Francisco. Should the primary computer fail, the backup can take over immediately. It is never used for off-line processing (Williamson et al., 1973).

The hot standby configuration is an expensive alternative. It requires two complete computers and extensive hardware and software to provide the automatic switchover.

The cold standby configuration is less complicated and therefore less expensive than the hot standby configuration. Although two full-sized computers are used for redundancy, less switching hardware is required for a cold standby system. Moreoever, the secondary computer need not be idle when the primary is in operation.

Santa Fe's Barstow Yard uses a cold standby system. PC minicomputers are used: One controls switching, another controls retarding, and the third is a supervisory backup. The backup computer takes data from the other two computers and compares them with certain established tolerance limits. If the input is found to be outside those limits, an alarm sounds which stops the hump operation. The program to control the process (switching or retarding) of the computer that has failed is then loaded onto the supervisory backup so that it can take over that function.

If there is no backup computer (or if both computers fail), it is necessary to revert to semiautomatic control. Southern Pacific's Eugene Yard, for example, has a relay backup system--pushbutton, relay routing, and "dial-a-speed" control of group retarders via a rheostat on the control console. Exit speeds of cars from the master retarder are still controlled automatically, using the cars' gross weights to determine the amount of retardation needed (WABCO, 1976).

In "smaller" YATS sites, installation of a minicomputer is planned to keep an on-line inventory concurrently with the YATS computer. With failure of the primary system, the computer can be used to print inventory cards in track-standing order. These cars can then be used for a manual card-PICL system.

Santa Fe's Barstow Yard has no backup MIS processor. If the computer fails, two high-speed terminals are used for data transmission to the COIN computer at systems central and to the PC computers.

5.4.4.2 Case Study Example.

Process Control--The simplest alternative when the PC computer at Potomac Yard fails is to return to the semiautomatic switching system. This does require, however, that both systems be kept in operating condition. In this case, a switch list must be printed from the MIS computer and manually entered. Humping exceptions must be manually entered via a CRT terminal to update the inventory.

Because of increased use of the hump with a PC computer, a manual backup may not be sufficient except for short periods and the cost of an inoperative hump is very high.

A failure of the PC computer stops the hump immediately, but its inoperation does not become critical for a number of minutes, a delay often experienced during the normal course of humping operations. Therefore, a cold standby computer configuration can be considered to be sufficient. A fail-safe configuration is probably best because all the expected functions of the PC computer are critical for yard operations and efficient data transmission to the MIS computer.

The secondary computer need not be idle when the primary compter is operative. Some thought must be given to which applications it can perform. When the primary PC computer fails, the standby computer cannot instantly take over the current PC functions because it does not have the switch list or the state variables of the current control process in its working storage. It must first unload its non-PC applications and then initiate the CPU with PC software and data tables.

MIS--Potomac Yard currently has an MIS computer configuration consisting of two computers. This system has been developed over a number of years, and a major consideration in the configuration selection was the software investment in the current system. One processor is used for external communication with tenants and communication with inquiry terminals. The other is used for the current MIS programs. These computers communicate through a direct processor-to-processor communications channel and share a number of disk drives. Figure 5-12 shows the current hardware configuration at Potomac Yard.

The MIS functions are not as time critical as the PC functions. A number of functions, such as tenant communications and receiving yard inventory, become critical in a number of hours rather than in number of minutes. In addition, other functions of the MIS, such as production of monthly management reports, are required infrequently. Other functions, such as some inquiries, although frequent, are not critical to yard operations.

Potomac Yard personnel believe that neither computer is used fully. Therefore, the availability of critical MIS functions can be improved by designing software to provide for fail-soft recovery for a processor failure. With this design, when one computer fails, the other continues to process the functions of both computers.

To properly design this fail-soft system, a priority of programs must be made. This list will be used to designate the programs that will continue to run when one processor fails. One order of priority might be:

- Switch list generation
- Receiving yard inventory
- Receive revised switch list
- Classification yard inventory
- Inbound program
- Receive advance consists
- On-line inquiries
- Departure yard inventory
- Outbound train consists
- Send advance consists
- Interchange reports
- MIS reports
- Shop reports.

5.4.5 Hardware Specifications

Hardware specifications define the hardware requirements for the yard. Potential hardware and system vendors use them to formulate a proposal to present to yard and railroad management.

The hardware specification is a document in addition to the functional specification developed during the system design phase and the software specification used for program development. For PC systems, hardware specifications are not as important as functional specifications because the specifics are determined by the vendor chosen to develop the total system. Hardware specifications are more important when a yard is developing or expanding an MIS computer facility that is independent from process control because the hardware needs to develop additional MIS applications are defined.

A hardware specification may be in two forms: a performance specification and an equipment specification.

5.4.5.1 Performance Specifications. The performance specification describes the volume of processing that will be required from the computer. Expected files and their sizes must be estimated to determine the size and type of storage that will be required. In addition, the number and size of each type of file must be determined, as well as the expected number of files that will be accessed at any one time. The frequency of access is also important for size determination and selection of medium type. Files that are often accessed, such as inventory files, must reside on a quickly accessible medium such as hard disk, whereas records that are kept only for historical purposes may be stored less expensively on removable disks or tape.

Determination of the size of the CPU is based on estimates of program sizes. The frequency of use and frequency of simultaneous use of programs must be known to estimate the size of the processor. Knowledge of the processing mode (whether a program is off-line or on-line and interactive) is also crucial for selecting a processor that can handle the load satisfactorily. The types of operating systems and use of memory and type of memory accesses are also determinants of hardware size and are part of a performance specification.

The volume of outputs is another performance requirement stated in the specification. Estimates of output volume, frequency, and growth are essential. Different methods of presentation, such as CRT terminals, printers, cards, external communications, and graphics displays, require different formatting, I/O channels, and response times. The volume and type of inputs constitute a performance requirement similiar to output. Again, estimates of the volume, frequency, method of presentation, and response time are required for performance specifications.

5.4.5.2 Equipment Specification. An equipment specification presents the actual hardware requirements. For a yard MIS upgrade or for PC system development, it may specifically identify interfaces to existing equipment. If an MIS computer is required for a new facility, complete hardware requirements beyond performance requirements may have been developed in the hardware study.

The first basic unit that is specified is the central processor. Attributes of the CPU are size, addressing, number of registers, cycle time, and instructions available. The amount of storage required is divided between mass and main memory. The operating system, programs, and data must be able to fit into main memory, so the total size is affected by the type of operating system. Virtual-memory operating systems require less main memory but require that mass memory be an extension of main memory. Cache memory speeds disk access when mass memory is used extensively in swapping users and programs in and out of core (main memory) in multiprogramming environments. Mass memory may extend beyond disk units when lower cost storage, such as tape, is required.

Communications requirements and I/O devices also must be specified. Required specifications for I/O devices such as CRT terminals and printers are type, speed, location, color or black and white, graphics capabilities, and print quality. The frequency and volume of external communications is also specified. The types of channels, speeds, priority, and media must be decided.

System software, especially the type of operating system, is an important consideration that affects selection of the size and type of all other hardware components. A number of operating systems and multiuser operating systems are available and each require different hardware. An additional consideration is to ensure that any purchased hardware is available with the software and languages that are expected to be required. This software should be available directly from and supported by the hardware vendor and in a standard ANSI version.

Additional hardware considerations that should be included in a more detailed specification are site requirements and limitations. The expected maintenance and reliability should also be specified, as well as future vendor support.

6.1 PROCUREMENT PROCESS

6.1.1 Request for Proposal

After software and hardware specifications have been developed and approved, RFPs may be issued to potential vendors. The RFP should cover the following areas:

- Functional specifications.
- Performance requirements.
- Hardware interfaces.
- Data storage requirements.
- Work load expansion capability.
- Location and site preparation.
- Security.
- Maintenance and availability.
- System backup/redundancy.
- Support by vendor, vendor responsibilities.
- Railroad responsibilities.

The RFP for a PC system should also include yard layout and design parameters, assumptions about track type and rolling stock, and existing field and PC equipment (see Exhibit 6-1).

After proposals have been received from vendors, they are evaluated for responsiveness, cost, availability, and the like. In the case of hardware-only purchases (where software will be developed in-house), plans for benchmark testing with responsive vendors are made.

6.1.2 Development of a Benchmark Test

A major technique for simulating an application on a computer is benchmark testing. This consists of running a mix of user programs that are representative of the predicted work load on a vendor's proposed computer system to validate system performance. This testing not only determines whether the computer has sufficient capacity for the planned applications, but also provides information on operating costs.

Benchmark tests are used more often with MIS computers because they indicate whether the computer selected has the capacity for expected growth. Benchmark tests are not often used to select PC systems because the application software provided is the major criterion for selecting a vendor. The development of a successful benchmark test requires several steps. The first is to develop a detailed definition and analysis of the work load for the new system. This must include near-term as well as longer term forecasts. In the second step, a representative mix of programs to be executed is chosen and the data to be processed are determined. The output of these steps must be carefully documented, together with the proposed techniques to test the operation and to measure response time, turnaround time, and transaction processing rates.

A formal test procedure is developed and presented to the vendors. This documentation helps to ensure that the vendors' software and hardware are evaluated as consistently as possible. Vendors, at this time, may request modifications and suggest equipment substitution. Clearly, all adjustments must be made before the actual test. Often, both a user and vendor representative measure and record timings. At the end of the demonstration, user representatives meet with the vendor to review the test results.

6.1.3 Negotiations and Contracting

For large-scale projects, a negotiating team is formed consisting of at least a data processing manager, a railroad lawyer (or counsel), and a railroad financial officer. Sometimes, one or more major users are also included. The selection of people for the team reflects the management structure of the railroad and/or yard.

The type of contract that is negotiated depends on a number of factors. Some of these are whether the hardware is to be purchased, leased, or rented and whether the vendor is to provide the hardware and/or software, maintenance services, and so on. Vendors have standard contract forms, which can be modified to cover the specific situation. Modifications are often specified by the project manager and reviewed by the railroad's lawyer or counsel. The contract should stipulate the requirements for hardware and software, the details of site requirements, delivery and installation, the documentation (such as hardware manuals) and maintenance to be supplied, the guarantees and limitations on the manufacturer's liability, the cost, and methods of payment. Ordinarily, the specifications documented in the RFP become a part of the final contract. In addition, the results of acceptance tests based on the performance requirements developed as part of the functional specifications should be documented.

EXHIBIT 6-1 RFP SYSTEM SPECIFICATIONS

A. General Overview

Goals of yard improvement
Expected traffic flow

B. Process Control

1. Yard layout and assumptions

   Existing and future yard
   - Track layout
   - Number of tracks, length
   - Distance from hump crest to tangent point
   - Hump profile (grade)
   - Humping speed.

   Assumptions about rolling stock--lengths, rolling resistance histograms, worst-case test (good following bad roller).

   Installation and switchover to new system.

2. Functional description of requirements

   Routing and switching
   - Functions
   - Performance--percentage of misswitches.

   Automatic speed control
   - Functions
   - Performance
     - Percentage of overspeed couplings
     - Percentage of cars stalled
     - Sensitivity of change in rolling resistance during a car's run
     - Percentage of catch-ups and cornering conflicts.

3. Alarms and reports

4. Hardware and communication requirements
   - Performance
   - Maintenance and reliability

C. Management Information System

1. Functional description

   Car record description
   Inventory
   Arrival procedures
   Humping and classification
   Departure procedures
   Inventory reports and inquiries
   Communications.

2. Hardware

   Performance
   Maintenance and reliability
   Backup.

3. Security and user access

4. Installation and switchover

D. Power and Operating Conditions

Power available
Environmental conditions
Site requirements and limitations (size, air conditioning, noise control, etc.).

E. Documentation and User Training

F. Vendor Responsibility

G. Railroad Responsibility

6.2 SOFTWARE DEVELOPMENT AND TESTING

6.2.1 In-House Development and Testing

If software is to be developed in-house rather than purchased from a vendor, detailed design can begin after the final specifications for software have been approved. This stage varies according to the completeness of the specifications but may include the production of module hierarchy charts indicating control flow among segments, descriptions of the functions performed by each segment as well as definition of all data inputs and outputs from each segment, and plans for testing the operation of each module. Coding should not begin until after the design stage has been completed, reviewed, and approved.

With detailed specifications and design, segments can be coded in parallel. Code reading, where the logic and format are checked by another programmer, is a reasonable and effective way to detect most errors--particularly when combined with a review by programmers whose work interfaces with the module and by the chief of the programming team.

Often errors are uncovered in a series of machine tests beginning with individual segment and module testing and then proceeding to integration testing, system testing, and acceptance testing (see Section 6.3).

6.2.2 Adaptation and Testing of Commercial Software Packages

Because of the variety of operational procedures and system requirements among railroads and classification yards, a packaged yard MIS or PC system will require some adaptation for any yard. When a PC system package is purchased from a vendor, adaptation, installation, documentation, and limited user training are usually included in the software cost. The programs should be tested with the vendor's test data set before any modifications are made.

Because an entire vendor package becomes available at one time, the testing does not follow the sequential pattern required for software that is developed in-house. Rather, the entire system can be tested concurrently.

6.3 DEVELOPMENT OF ACCEPTANCE TEST PLAN, SPECIFICATIONS, PROCEDURES, AND TEST DATA

Acceptance testing, which is the final step before system implementation, requires careful planning and scheduling. Its components are preparation of test specifications (to be documented in the contract if the system is developed by an outside contractor), test procedures, and development of test data.

Planning for the acceptance test is one of the most important steps in the development process. The acceptance test plan should be developed concurrently with the software system. A major part of the planning process is to identify individual test tasks. This is accomplished by subdividing the acceptance test into functional components similar to those in the logical model created in the analysis phase of systems design (5.2.2). For example, for the function of verifying and updating an inbound consist, system planners must review existing and proposed clerical procedures to create and document test procedures for the new software programs. Obtaining reliable test data can be very time consuming but is worthwhile because, in addition to being used for acceptance testing, the data can be saved and used to test future system modifications.

An example of acceptance testing of software is as follows. A test inbound consist file would be created with intentional errors in data and car order. Incoming waybills, ACI data, or videotape records would be used in the test to modify the existing record. The modified records would then be compared against a prepared file of corrected consist records.

Acceptance testing of new hardware must also be conducted. Equipment tests are first performed by the manufacturer, usually on the user's premises after installation. Plans for the final test must include provision for complete user involvement. Procedures must be developed to report and act on discrepancies that are encountered.

6.4 SYSTEM DOCUMENTATION

System documentation is developed at various stages in the systems design phase (Chapter 5) and in the system acquisition and implementation phase and continues on into the operations and management phase (Chapter 7). The functional specifications (discussed in Section 5.2) are developed into system specifications that give detailed design for the software programs, as discussed in Section 5.3.4. The next set of documents to be prepared cover the detailed structure of the software, which includes preparing flowcharts and program module documentation.

The final two documents to be developed are manuals--one for operations and one for users. The operations manual should be prepared before system testing and installation, and the user manual shold be completed before user training begins.

Computer operation manuals describe: (1) daily computer runs, (2) who provides the input, (3) who obtains the output, (4) what tapes and disks are needed and how they are identified, (5) what files are to be used and how long each should be saved, and (6) how long each program runs and how much memory it uses. The computer operation manual also contains information on backup and restoration of files. PC computer operation manuals require few procedures because PC computers are usually dedicated to real-time process control and user and field equipment communication. Procedures are often limited to maintenance and file backup and restoration.

User manuals contain procedures for preparing input and for checking and using output. They contain samples of all forms and reports, with detailed instructions for their handling and use. They state how to add, remove, or change records.

As the systems development cycle proceeds, documentation in the later stages is based to a great extent on that developed in the earlier stages. For example, if functional specifications are well prepared, they should provide most of the information required for the user manual.

The extent of the documentation can vary, depending on the size and complexity of the systems to be installed and on management philosophy. For example, in its software development guidelines, the federal government specifies up to 10 documents for use in very large projects (U.S. Department of Commerce, 1976). A data requirements document is separate from the functional specifications document. System specifications in the government system may comprise three separate documents--one covering the system/subsystem, one covering the program, and one covering the data base. In addition to a user manual and an operation manual, a program maintenance manual is sometimes produced, as is the separate formalized documentation of a test plan and at least an informal test analysis.

6.5 USER TRAINING

Part of the process of installing a new or modified system is training or retraining the users in the yard, the DP operators, and programmers. The provision for training depends on a number of factors, such as whether the system is new or modified (and the extent of the modification) and whether the software has been developed in-house or purchased.

Some of the user training can be a by-product of the acquisition process. For example, the functional specifications are promulgated to advise the users of how the new (or modified) system will affect them, and, as noted in Section 5.1.1, staff members as well as user supervisors are expected to work with the project team. Small modifications to existing MIS programs require only in-house training. Training and new procedure documentation should be provided for each employee who may be required to use the new MIS functions. Training materials can be obtained by modifying the functional specification and system design documentation. If additional hardware or operating system programs are purchased, the supplier will in most cases offer preprepared documentation. If a complete custom software package, such as PC applications, has been purchased, training should be provided for a few personnel who, using the vendor's training documentation and procedures, can train others in the yard.

A training plan should be carefully documented, indicating the users who require the training, the content of the training, and the time required for the training. Such planning helps to ensure that training is completed by the time the new system becomes operational.

6.6 SITE PREPARATION

Site preparation considerations include not only the layout of the new system, but also the flooring, environmental specifications, power requirements, cabling and grounding, and fire protection, security, media storage, and safety. The contract for the purchase of hardware should address many of the requirements for the site, such as written environmental specifications (including temperature control, humidity, and contaminant level of air) for optimum operation.

Hardware may be required to interface the new computer with the existing computer systems or field equipment. Interface and communications requirements will be defined by the vendor but purchase and installation may be the responsibility of railroad purchasing. Establishing the interface to the field equipment may be the most complicated because field equipment requires an analog-to-digital and digital-to-analog conversion for communication.

The addition of a PC system to an existing yard may require more field equipment. Additional presence detectors and/or wheel counters may be required for car tracking and counting and for rollability estimation. Distance-to-couple track circuits may also be installed in classification tracks to return distance-to-couple information to the speed control function, the track overflow alarm routine, and MIS inventory records. This equipment can be installed track by track in a phased cutover by track group. This will ensure that a minimum of cars are affected by the closure of classification tracks. Each new device will in turn be connected to the PC computer.

Vendors can be helpful in providing assistance during the site selection, design, and preparation phases. Because the purchase of a new system may require erection of new buildings or extensive renovations of old buildings, detailed planning and scheduling are mandatory. In deciding the physical placement of equipment, for example, planners must take operational and maintenance considerations into account. Clearly, the possibilities for future expansion must also be addressed. Even the ordering and delivery of supplies such as magnetic disks can be critical in effecting an orderly system transition and on-time project performance.

If in-house expertise is not available or additional expertise beyond that of the vendor is required for site design and preparation, an outside consultant or engineering firm can be contracted to provide assistance. In addition to, or instead of, a consulting arrangement, a facilities maintenance engineer may be hired to be responsible for maintenance and operations after acceptance testing is completed at larger EDP facilities.

6.7 SYSTEM CONVERSION

Several methods of effecting system conversion are available. One is to use a parallel operation, whereby both the new and the old system are run concurrently while outputs of the new system and old system are compared to increase assurance that the new system is performing properly. At the other extreme, a complete cutover to the new system is made after extensive system tests. This method has risks but, if it is well managed, can be the least expensive of the available alternatives. These methods assume that both old and new systems perform similar functions and can be compared.

By combining these two methods in a "phased" cutover, only parts of the system are converted at a time. Some parts may be converted in a parallel mode, whereas others may be converted without recourse, depending on the complexity and other factors that affect the individual components.

At times, emulation or simulation may be used to assist in converting from one system to another. These processes enable the cutover to proceed more slowly than would otherwise be possible.

For safe and efficient conversion from an existing PC system (relay base or other) to a PC computer, parallel operations are suggested. In turn, the speed control function and automatic switching and routing function must be installed. For example, incoming signals are tapped from the input junction of the existing relay PC system and passed through an analog-to-digital converter to the PC computer. In each case, the PC computer is connected in parallel with incoming field sensors and equipment signals. These signals are used to drive simulated output from the computer. The simulated output signals from the PC computer are compared with actual and expected output signals as a test of the new system. Once the simulation shows that the PC computer works correctly for a particular part of the yard, actual tests can be run. The existing relay PC system is used as a safety backup while the new computer system is being tested. Once parallel simulation and actual tests are completed, the existing relay-based PC system can be phased out by disconnecting it from the field communications.

6.8 ACCEPTANCE TEST AND SYSTEM INSTALLATION

For both new and modified facilities, software and hardware should be thoroughly tested for acceptance before they are used for actual full-time operations. All acceptance tests are performed in accordance with the test plan and specifications in the contract. Test results should be documented, as should corrective measures that have been taken. User acceptance and approval should also be formalized.

Testing PC software before system installation is often infeasible when it is being added to an operating yard. A gradual phased conversion can be used in the acceptance test. As each part of the system is tested and successfully cut over, acceptance is acknowledged by the yard.

7.1 COMPUTER PERSONNEL ORGANIZATION

The organization of personnel for computer systems operations and management depends, among other factors, on whether the system is centralized, regionalized, or decentralized. It also depends on the number and size of the yards in the system; whether the system covers MIS or PC applications, or both; and the complexity of the new system.

The EDP organizational policy varies from railroad to railroad. Larger railroads with a sizable network of yards and rail links tend to centralize their EDP functions at the corporate headquarters (or regional headquarters in a few cases). The yard EDP personnel in these railroads are either under the control of the corporate data processing vice president (if they are assigned to yard MIS functions) or the corporate vice president for engineering and signaling (if they are assigned to yard PC functions). The yard EDP personnel under corporate control are primarily responsible for systems design and implementation, and their objective is to maintain corporatewide standards and eliminate unnecessary duplication of effort between yards. However, other EDP personnel also are users of the systems, responsible for carrying out the yard EDP operation as directed by the yard superintendent.

Under a decentralized arrangement, the yard EDP functions are more independent than under the centralized arrangement, and the EDP personnel located in the yard are under the direct control of the yard superintendent. They also serve as a liaison between the corporate EDP and the yard.

In railroad companies whose business is primarily derived from yard operations, the yard superintendents often have complete control over the yard EDP functions, including computer systems design, systems development, equipment selection, and operational control. The corporate EDP personnel, in this case, serve as consultants and liaisons between the yard EDP and corporate EDP departments.

Regardless of whether they are performed by the central EDP personnel or by the field EDP personnel, the EDP functions eventually affect the operation of the entire railroad. Consequently, the EDP groups in many railroad companies are under the scrutiny and guidance of a companywide EDP advisory committee. The members of the committee usually represent a cross-section of the company, including the executive office, marketing, finance and accounting, research and planning, signal and engineering, train control and operations, and yard management. Although the EDP advisory committee may not have direct administrative authority over the EDP operating organizations, its opinions and recommendations to the top management may significantly affect the direction and scope of the EDP activities.

The responsibility and authority for EDP functions between the yard and the company central facility may vary from company to company. In general, the following functions are viewed as the primary responsibilities of the central EDP department in a centralized organization:

- Establish EDP goals and objectives.
- Assume responsibility for corporatewide design and development of EDP systems.
- Develop and maintain common standards, policies, and rules.
- Assume responsibility for centralization of hiring, training, and cross-utilization of capabilities.
- Provide applications analysis and coordination and software design, development, and maintenance.
- Select and procure hardware.
- Establish data communication interfaces.
- Provide computer operations and services at the central EDP location--for example, waybill processing, customer records, accounts payable, payroll, general accounting, personnel record processing, inventory and property accounting, research and engineering, and systemwide MIS.
- Balance demand and supply in EDP work load among yards.
- Maintain proper interface between yards and between yards and central.
- Provide EDP consulting services for yards.

Assuming that some computer equipment (from a "smart" terminal to a full-fledged computer center) is installed in the yard, the responsibility of the yard personnel would then encompass:

- Day-to-day operational functions.
- Hardware maintenance.
- Input data acquisition and preparation.

- Output data dissemination and utilization.

- Coordination with central EDP and other yards.

- Local modifications from corporatewide standard packages.

- Provision of local operational expertise to central staff in EDP design and development work.

In some areas, the responsibility and authority are not clearly divided between the central and the field. Some examples are:

- Designing and programming of yard management decision models, for example, dynamic track assignment, crew dispatching, engine dispatching.

- Modification of parameter values and weights of objective functions in these decision models.

- Modification of standard software packages to make them more compatible with local applications.

- Maintenance work on yard computer software.

In most railroads, the MIS function is the responsibility of an EDP department. In most larger organizations, an EDP operations manager reports to the data processing manager, who has other management responsibilities, such as those for systems analysis and programming. The operations manager, particularly in a centralized system, is likely to have responsibilities other than those related to the classification yard; for example, the manager may be responsible for a systemwide MIS computer operation.

Usually, the EDP operations manager has responsibility for the MIS computer facilities as well as the hardware, software, operating procedures, and data files. The overall PC responsibility is usually assigned to the company-level signal and/or engineering organizations. The signal division frequently handles PC hardware selection and procurement, software development or procurement, operational design, and so forth. However, the day-to-day computerized PC operation in the yard is controlled by the yard management. In some cases, the yard is also responsible for computer programming, wiring of hardware interfaces, and the development of control and data reduction logics peculiar to that yard. The responsibility for PC software and hardware maintenance may be contracted to the system vendor. In other cases, on-site responsibility for the hardware may lie with the EDP department because expertise is readily available.

7.2 RECRUITMENT AND TRAINING OF PROFESSIONALS

The types of computer professionals that are required to run the EDP operations of a railroad company include the following:

- Computer center administrative specialists.

- Software system designers.

- System programmers.

- Applications analysts.

- Applications programmers.

- Applications consultants.

- Computer operators.

- Maintenance specialists.

- Key entry operators.

- Process control/real-time application specialists.

- Software librarian.

- Computer architecture specialists.

- Data communications specialists.

Because the EDP field is growing so rapidly, many railroad companies are having difficulty in recruiting and retaining qualified computer personnel; this is even more difficult for classification yards. The major reasons apparently are the following:

- Labor union agreements and company personnel policy tend to limit the types of computer personnel who can work in a classification yard. Therefore, the EDP job classifications available in a yard are not attractive to the regular job seekers.

- With certain exceptions, the remoteness of the classification yard from urban centers, where most computer people are trained and employed, tends to limit the yard recruiters' access to the EDP personnel supplier market.

However, this trend at classification yards may very well change in the future as more and more yards install sophisticated computer systems. Thus, more interesting and challenging work in the future may be offered in the yards rather than in the corporate headquarters.

One solution to the recruitment problem used by many railroad companies is to select potential EDP candidates from the ranks of career railroad employees occupying non-EDP positions and train them to be computer specialists. Developing EDP expertise in a person with a railroad background generally takes less time than familiarizing an EDP professional with railroad operations. Moreover, an EDP professional from the railroad is more likely to stay with the company because he or she may already have established a career within the company.

The training of yard EDP personnel is usually divided into two stages. During the first stage, trainees receive formal training and orientation designed and administered by the corporate EDP organizations. The purpose of this stage is to ensure that the trainees understand corporate-wide standards and procedures. In the second stage, the new EDP professionals receive specific training in the yard on all aspects of yard operations and some in-depth on-the-job training in yard EDP operations.

7.3 OPERATING PROCEDURES

For the efficient management of a large yard EDP facility, complete operating procedures must be developed for job flow, data preparation, data entry, computer operations, and quality assurance. Establishment of file inventory procedures and procedures for file retention and backup is also important. Because magnetic tapes and disk packs can easily be damaged or destroyed, the operations manager must establish retention and backup standards and procedures.

The trend has been away from "open shops" where programmers, computer users, and others can wander freely in and out of computer facilities. As computer operations have become larger and more complex, many computer centers are adopting a semiclosed or closed site. A closed shop is more efficient because computer personnel are protected from outside interruptions, and security is very important. Thus, the closed shop is more cost-effective, more productive, more accurate, and more secure. Other advantages include a clear-cut division between the end of program design and beginning of operations, with a concomitant requirement for more complete and clear documentation of programs.

7.4 SYSTEM MAINTENANCE

System maintenance encompasses both hardware and software. Contracts often specify vendor responsibilities for hardware maintenance. Railroads have tended to use this option because PC computers and decentralized MIS computers are usually small and often at remote locations incapable of supporting a full-time maintenance technician. Specialized vendor maintenance personnel are often required to repair problems with field equipment and hardware interfaces unique to PC systems. Sometimes, it is advantageous to contract for third-party maintenance--particularly if hardware has been purchased from more than one vendor. Nonetheless, most larger centralized EDP facilities have at least some railroad personnel who are capable of handling minor maintenance problems.

All facilities require a preventive maintenance program. This is particularly important for PC computers because their failure impedes all yard operations. Often, weekly scheduled maintenance is arranged to be performed by the vendor--either as a free service or for a fee. Manuals provided by the vendors specify what maintenance is required. Written procedures should be generated internally that specify what to do if the system fails. Such documentation may include lists of causes and solutions for problems that may be encountered.

Many hardware vendors can provide diagnostic programs to isolate a problem before a service engineer is called. This is particularly valuable at remote yards. If a programmer or user first attempts to isolate the problem and can obtain advice by telephone from the vendor, many unnecessary and costly service trips can be avoided.

Good maintenance practices dictate that a log be kept of systems failures, specifying the date, equipment, problem, cause, and solution. This log can assist in identifying causes of future maintenance problems and also identify equipment that has frequent maintenance problems.

Large railroads with complex software programs may also have detailed maintenance manuals that include program descriptions, as well as descriptions of the hardware, the support software, and the data base. Maintenance procedures are specified that include programming conventions, verification procedures, error correction procedures, special maintenance procedures, and lists of the programs and flowcharts. Such manuals are particularly valuable in facilities where turnover rates are high.

7.5 METHOD OF DAY-TO-DAY MONITORING AND CONTROL

Day-to-day monitoring and control of an EDP facility is the responsibility of the operations manager or equivalent. Many methods are used to successfully accomplish these functions, including using various performance measures, job accounting, and hardware and software monitors.

Performance measures ideally are incorporated into the system during the design phase of the project, with special attention given to those aspects of the system that most affect productivity. The most universal performance measures used to assess on-line systems are:

- Terminal response time (time from message transmission to receipt of result).
- System availability (usually measured as total scheduled availability less downtime divided by total scheduled availability).
- Transaction volume (number of transactions in a specified time period).

Although job accounting (log-in time for specific jobs) is effective for evaluating performance of batch processing systems, it is seldom used to measure the performance of on-line systems, and most new railroad information systems are on-line systems.

Hardware and software monitors are often recommended as performance evaluators for certain operating systems or applications. However, except for terminal response time, hardware monitors are not equipped to measure application performance, which is of most importance in classification yards. In addition, hardware monitors are expensive and require people with highly specialized skills to analyze the data. Stopwatches, although not as accurate as hardware monitors, can often provide sufficiently accurate data to measure terminal response time.

Software monitors are not as expensive as hardware monitors, and their use is much more widespread than that of hardware monitors. Software monitors can be used to collect data on the performance of on-line system functions and applications.

Performance monitoring is particularly important because yard applications must interface with real-time events. One measure of the performance of an MIS computer is whether inventory transaction processing can keep pace with actual events. The lag between physical events and the change in the MIS inventory data base reflect the load on the computer if all input has been made promptly. Occasionally, MIS processing delays may be acceptable if humping does not depend on inventory information. On the other hand, promptness is much more critical for PC applications. PC commands for speed control must be completed in milliseconds for the proper operation of field equipment.

Allman, W. P. "Computer Assistance for Real-Time Planning of Yard Operations." Bulletin IRCA: Cybernetics and Electronics on the Railways. International Railway Congress Association: Brussels, Belgium, February 1968, pp. 75-79.

Amdahl, L. "The Technology Swirl." Datamation, vol. 24, no. 12, November 1978, pp. 18-20.

Anderson, L. E. "Union Pacific's Terminal Information System." Papers and Committee Reports Presented at the 1977 Annual Meeting. Association of American Railroads, Data Systems Division: Washington, D.C., 1977, pp. 49-50.

Auerbach Publishers, Inc. "Structured Analysis." Information Systems Analysis Methods and Tools. Publ. No. 32-04-03. Auerbach Publishers, Inc.: Pennsauken, New Jersey, 1979.

__________. Auerbach on the Supercomputers and Array Processors. Auerbach Publishers: Pennsauken, New Jersey, 1981.

__________. Data Processing Management. Vol. 1, Auerbach Information Management Series. Auerbach Publishers, Inc.: Pennsauken, New Jersey.

__________. Systems Development Management. Vol. 2, Auerbach Information Management Series. Auerbach Publishers, Inc.: Pennsauken, New Jersey.

__________. Data Center Operations Management. Vol. 3, Auerbach Information Management Series. Auerbach Publishers, Inc.: Pennsauken, New Jersey.

Auerbach Reporter. "Computer Technology Forecast--Part 1." Auerbach Reporter, Computer Technology Edition, February 1978.

__________. "A Practical Guide to Distributed Processing." Auerbach Reporter, Computer Technology Edition, September 1979.

__________. "IBM Revolutionizes the Computer Industry." Auerbach Reporter, Computer Technology Edition, April 1979.

__________. "Ethnotronics--The Shape of Things to Come?" Auerbach Reporter, Computer Technology Edition, January 1980.

Bard, Y. "The VM/370 Performance Predictor." ACM Computing Surveys, vol. 10, no. 3, September 1978.

Becker, H. B. "Let's Put Information Networks into Perspective." Datamation, vol. 24, no. 3, March 1978, pp. 81-86.

Blumenthal, M. "Vendors: 4300 Means User Relief." Computer World, February 5, 1979.

Boeing Computer Services, Inc. Making It Count. Vols. I and II, Boeing Computer Services, Inc., 1974.

Business Week. "No Boom for Bubble Memory." Business Week, May 4, 1981, pp. 152-153.

Carlin, B. J. "A Status Report on Waybill Data Exchange." Papers and Committee Reports Presented at the 1979 Annual Meeting. Association of American Railroads, Data Systems Division: Washington, D.C., 1980.

Chamberlain, H. L. "Application of Mini-Computers on the Missouri Pacific." Bulletin 643, American Railway Engineering Association: Chicago, Illinois, 1973.

Champine, G. A. "Six Approaches to Distributed Data Bases." Datamation, vol. 23, no. 5, May 1977, pp. 69-72.

__________. "What Makes a System Reliable?" Datamation, vol. 24, no. 9, September 1978, pp. 195-206.

Coleman, S. "SEL Brings Out 'Most Powerful' 32-Bit Mini." Computerworld, May 11, 1981.

Computing Surveys, vol. 10, no. 3. Association for Computing Machinery, Inc.: New York, September 1978.

Computer, vol. 13, no. 4. IEEE Computer Society, Long Beach, California, April 1980.

Couger, J. D., and McFadden, F. R. Introduction to Computer Based Information Systems. John Wiley and Sons, Inc.: New York, 1975.

Coughlin, V. "Selecting an Intelligent PC." Design Engineering, vol. 52, no. 4, April 1981.

Datamation. "News in Perspective, Refresh Your Memory." Datamation, June 1981, p. 78.

DeIvernois, P. J. "Evolution of Automatic Controls in the Classification Terminal Area." Paper presented at IEEE Annual Convention, March 21, 1972, New York; reprinted in Bulletin 282, WABCO: Swissvale, Pennsylvania.

Digital Equipment Corporation. Program Demonstration: The Missouri Pacific's Yard and Terminal System (YATS). Digital Equipment Corporation, 1975.

Dikinson, J. Union Railroad--Computerized Crew Management System, A System Overview. Bessemer and Lake Erie Railroad, October 1979.

DiPaola, J. J. "The Distributed Computer System in Classification Yards." Bulletin 643, American Railway Engineering Association: Chicago, Illinois, 1973.

Ein-Dor, P. "A Dynamic Approach to Selecting Computers." Datamation, vol. 23, no. 6, June 1977, pp. 103-108.

Flynn, M. "VLSI: The Next Computer Generation." The Stanford Engineer, Fall/Winter 1979, pp. 19-24.

Gabet, J. M. "VLSI: The Impact Grows." Datamation, vol. 25, no. 7, June 1979, pp. 108-112.

Gildersleeve, T. R. Data Processing Project Management. Van Nostrand Reinhold Company: New York, 1974.

GRS Co. "GRS Automatic Switching for Car Retarder Yards, Circuit Description and Maintenance." Pamphlet 933, General Railway Signal Co.: Rochester, New York, December 1957.

___________. "GRS Distributed Computer Systems." GRS Development D21.0200, General Railway Signal Company: Rochester, New York, October 1971.

___________. "CABMATIC: GRS Automatic Cab Signal System." Pamphlet 1582, General Railway Signal Company: Rochester, New York, May 1977.

___________. "Class-Matic Mark II, Total Automatic Freight Classification." Bulletin 206, General Railway Signal Company: Rochester, New York.

Graziani, J. M. "Conceptual Approach to Yard Systems." Papers and Committee Reports Presented at the 1977 Annual Meeting. Association of American Railroads, Data Systems Division: Washington, D.C., 1977, pp. 38-48.

Hall, I. B. "ICG's Interchange Continuity System (ICS)." Papers and Committee Reports Presented at the 1979 Annual Meeting. Association of American Railroads, Data Systems Division: Washington, D.C., 1980, pp. 196-198.

Hartman, W., Matthes, H., and Proeme, A. Management Information Systems Handbook. McGraw-Hill Book Co.: New York, 1968.

Hermann, D. J. "SCERT, A Computer Evaluation Tool." Datamation, vol. 13, no. 2, February 1967.

Hodges, D. A. "Microelectronic Memories." Scientific American, vol. 237, no. 3, September 1977, pp. 130-145.

Huskey, H. "Computer Technology." Annual Review of Information Science and Technology, vol. 5. Encyclopedia Britannica, Inc.: Chicago, Illnois, 1970.

Isenberg, E. J. "Elimination of Paper Interchange Reports." Papers and Committee Reports Presented at the 1977 Annual Meeting. Association of American Railroads, Data Systems Division: Washington, D.C., 1977, pp. 73-74.

Johnson, D. L., and Steiner, F. T. "Process Control Computer Systems for Railroad Applications: Site Preparation and Maintenance Considerations." Bulletin 327, WABCO, Union Switch and Signal Division: Swissvale, Pennsylvania, May 1978.

Karvwatt, R. C. "The Use of Digital Computers in Yard Switching and Terminal Operations." Bulletin IRCA: Cybernetics and Electronics on the Railways. International Railway Congress Association: Brussels, Belgium, February 1968, pp. 80-84.

Kenney, D. P. Minicomputers. Amacom: New York, 1978.

Kurka, F. "Judging the Superminis on a System Basis Is Imperative for Full Use." Computer Business News, December 25, 1978.

Laden, H. N. "Process Automation and Management Control in Yards and Terminals." Bulletin IRCA: Cybernetics and Electronics on the Railways. International Railway Congress Association: Brussels, Belgium, February 1968, pp. 69-75.

Lay, N. J. "Grand Trunk Western Railroad Co. Train and Yard Planning." Papers and Committee Reports Presented at the 1979 Annual Meeting. Associaton of American Railroads, Data Systems Division: Washington, D.C., 1980, pp. 106-134.

Lister, A. M. Fundamentals of Operating Systems. The Macmillan Press, Ltd.: London, 1975.

Long, P. L. "Computer Technology--An Update." Annual Review of Information Science and Technology, vol. 12. Encyclopedia Britannica, Inc.: Chicago, Illinois, 1977.

Lundell, E. D. "Forecast for the 80's: IBM Control Systems with More Functions." Computer World, February 19, 1979.

Lynch, P. A. "Small Yard Support System." Papers and Committee Reports Presented at the 1977 Annual Meeting. Association of American Railroads, Data Systems Division: Washington, D.C., 1977, pp. 70-72.

Martin, E. A. "Programmable Controllers--An Alternative to Computers for Process Control." AIIE News, Computer and Information Systems, vol. 15, no. 4. American Institute of Industrial Engineers, Inc.: Norcross, Georgia, Spring 1981.

Martin, J. Programming Real-Time Computer Systems. Prentice-Hall, Inc.: Englewood Cliffs, New Jersey, 1965.

__________. Design of Real-Time Computer Systems. Prentice-Hall, Inc.: Englewood Cliffs, New Jersey, 1967.

__________. Principles of Data-Base Management. Prentice-Hall, Inc.: Englewood Cliffs, New Jersey, 1976.

Martin, J. R., and Durand, S. C. "Implementation of a Distributed Computer Complex in Sheffield Yard." Fourth International Symposium on Railroad Cybernetics, Washington, D.C., April 21-26. Associaton of American Railroads: Washington, D.C., 1974, pp. 30-41.

McClellan, T. C. "Frisco's IBM 3790 Distributed Processing Control Concept." Papers and Committee Reports Presented at the 1977 Annual Meeting. Association of American Railroads, Data Systems Division: Washington, D.C., 1977, pp. 51-56.

McGlynn, D. R. Distributed Processing and Data Communications. John Wiley and Sons, Inc.: New York, 1978.

Myers, G. J. Advances in Computer Architecture. John Wiley and Sons, Inc.: New York, 1978.

Noyce, R. N. "Microelectronics." Scientific American, vol. 237, no. 3, September 1977, pp. 63-69.

Page-Jones, M. The Practical Guide to Structured Systems Design. Yourdon Press: New York, 1980.

Petracek, J. J. et al. Railroad Classification Yard Technology: A Survey and Assessment. Prepared under Contract DOT-TSC-968, U.S. Dept. of Transportation, Transportation Systems Center. SRI International: Menlo Park, California, July 1976.

Progressive Railroading. "West Colton...Southern Pacific's New Pivotal Yard." Progressive Railroading, September-October 1973, pp. 58-64.

__________. "Communications/Signaling/Computers 'Make' Barstow." Progressive Railroading, July 1976, pp. 59-60.

__________. "Strawberry Yard--One for Five in Louisville." Progressive Railroading, October 1977.

Railway Gazette. "Marshalling Yard Control Using Punched Cards." Railway Gazette, January 15, 1965.

Reynolds, C. H. "Issues in Centralization." Datamation, vol. 23, no. 3, March 1977, pp. 91-98.

Sargent, G. A. "Railroad Freight Movement--Some New Concepts and Current Plans." Transportation Engineering Journal, vol. 100, no. TE2, March 1974, pp. 475-487.

Sayers, A. Operating Systems Survey. Auerbach Publishers: Pennsauken, New Jersey, 1971.

Scherr, A. An Analysis of Time-Shared Computer Systems. Massachusetts Institute of Technology Press: Cambridge, Massachusetts, 1967.

__________. "Distributed Data Processing." IBM Systems Journal, vol. 17, no. 4, 1978.

Schiller, D. E. "System Capacity and Performance Evaluation." IBM Systems Journal, vol. 19, no. 1, 1980.

Schultz, B. "The Year of the Superminis." Computer World, February 12, 1979.

Shaw, J. C., and Atkins, W. Managing Computer System Projects. McGraw-Hill Book Company: New York, 1970.

Southern Pacific. "This is Southern Pacific." Southern Pacific Bulletin, vol. 57, no. 5, June 1973.

Stein, D. L. "Price/Performance, Semiconductors and the Future." Datamation, vol. 25, no. 13, November 25, 1979, pp. 14-20.

Steiner, F. T., and Johnson, D. L. "Process Control Computer Systems for Railroad Applications, Site Preparation and Maintenance Considerations." Bulletin 327, WABCO, Union Switch and Signal Division: Swissvale, Pennsylvania, May 1978.

Thurber, K. Large Scale Computer Architecture. Hayden Book Co.: Rochelle Park, New Jersey, 1976.

Townsend, D. F. "Systems Analysis: Key to the Future." Datamation, vol. 26, no. 10, October 1980, pp. 145-148.

U.S. Department of Commerce. Guidelines for Benchmarking ADP Systems in the Competitive Procurement Environment. Federal Information Processing Standards Publ. 42-1, National Bureau of Standards: Washington, D.C., May 15, 1977.

U.S. Department of Commerce. Guidelines for Documentation of Computer Programs and Automated Data Systems. Federal Information Processing Standards, Publ. 38, National Bureau of Standards: Washington, D.C., February 15, 1976.

Vacroux, A. G. "Microcomputers." Scientific American, vol. 232, no. 5, May 1975, pp. 32-40.

Van Rensselaer, C. "Centralize?, Decentralize?, Distribute?" Datamation, vol. 25, no. 4, April 1979, pp. 88-97.

Wellman, L. E. "Union Pacific-Santa Fe Paperless Interchange." Papers and Committee Reports Presented at the 1977 Annual Meeting. Association of American Railroads, Data Systems Division: Washington, D.C., 1977, pp. 78-79.

Wert, C. F., and McGlumphy, G. F. "Total Terminal Control." Paper presented to the Annual Meeting of Communications and Signal Section of the Association of American Railroads, Dallas, Texas, October 1-3, 1968; reprinted in Bulletin 240, WABCO: Swissvale, Pennsylvania.

Westinghouse Air Brake Company. "Computers in Railroad Control, The Progress and the Promise." Bulletin 302, WABCO: Swissvale, Pennsylvania, 1976.

Whitehead, A. C. "Southern's Sheffield and Savannah Yard Control Systems." Papers and Committee Reports Presented at the 1977 Annual Meeting. Association of American Railroads, Data Systems Division: Washington, D.C., 1977, pp. 57-63.

Williamson, W. V., Nagel, R. C., and Young, F. E. "West Colton Yard." Paper presented to Annual Meeting of Communications and Signal Section of the Association of American Railroads, Data Systems Division: New York, October 3, 1973.

Wong, P. J., Elliott, C. V., and Hathorne, M. R. Demonstration of Dynamic Track Assignment Program. Phase 1 report for the Association of American Railroads, Freight Car Utilization Program. SRI International: Menlo Park, California, June 1979.

Yourdon, E. "Different Concepts of Reliability." Modern Data, January 1972, pp. 36-42.

__________. Design of On-Line Computer Systems. Prentice-Hall, Inc.: Englewood Cliffs, New Jersey, 1972.

This appendix,* which describes the physical and operational characteristics of classification yards, provides the necessary background for understanding the operational context of yard computer systems. Those already familiar with yards may wish to omit this discussion. It is provided mainly for readers who have a computer background and little experience in yard operations.

This is a general overview of the information flow relating to the movement of freight cars through a typical classification yard. Certain operational details may vary significantly between yards. Part of this variance results from the difference in the systemwide information-processing procedures that the various railroad companies use. There can even be dramatic differences in the information-processing procedures in other yards operated by the same railroad company.

A.1 PHYSICAL DESCRIPTION OF CLASSIFICATION YARDS

A flat yard generally consists of a series of tracks connected by a ladder track and switching lead, as shown in Figure A-1.** Most flat yards use the same tracks for receiving, classifying, and dispatching trains, although many have separate receiving and/or departure tracks.

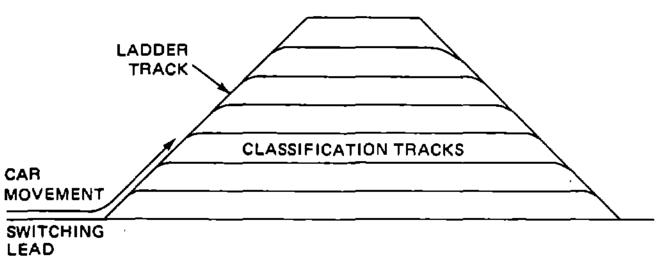


FIGURE A-1. EXAMPLE OF A FLAT YARD TRACK CONFIGURATION

The car-sorting process requires that a group of cars be pulled out to the switch lead, where the switch engine accelerates quickly toward the yard and then decelerates. Just before the deceleration, a car or group of cars is uncoupled and the deceleration of the switch engine and the cars coupled to it causes one or more of the uncoupled cars to separate from the rest. This procedure is called giving the cars a "kick."

The switch engine generally continues kicking cars toward the classification tracks until it reaches the ladder track, at which point it pulls the remaining cars back along the switch lead and resumes the process. The cars and groups of cars that have been kicked travel along the switch lead and ladder track until they are switched onto the appropriate classification track. Switches in most flat yards are generally thrown manually. To improve operations, the grades of flat yards are often somewhat saucer shaped so that the cars tend to accumulate in the center of the yard when switching is being done from both ends of the yard. Such gradients also reduce the frequency of cars stopping short on the ladder track or classification track.

Hump yards can classify a large volume of cars more efficiently than flat yards. Typically, a hump yard has separate receiving, classification, and departure subyards; Figure A-2 shows an in-line yard configuration. For classification, a yard engine takes a group of cars from the receiving yard and pushes them over a raised portion of track called the hump. Cars are uncoupled at the hump crest and begin to accelerrate down the hump grade, thereby separating from the yard engine and the remaining cars. As the cars roll down the hump grade, braking devices called retarders (see Figure A-2) control the speed of the cars, and the appropriate switches are thrown to route the cars into the designated classification tracks. Master, group, and tangent-point retarders are shown in Figure A-2, but many other types of retarder configurations exist, the most common of which is the combination of only a master retarder and group retarders.

A.2 CLASSIFICATION YARD OPERATIONS

The operations in a classification yard are keyed to the processes involved in receiving and breaking up inbound trains, classifying or sorting cars, and making up outbound trains. Figure A-3 depicts most of the major processes involved in moving a car through a classification yard. This flowchart indicates that the individual freight cars usually arrive with other cars on road-haul trains, transfer trains, or industrial drags and are placed on one of the yard tracks, if one is available. Most hump yards and a number of flat yards dedicate certain tracks or groups of tracks for receiving inbound trains. If a yard does not

*The material for this appendix draws liberally from information contained in Railroad Classification Yard Technology: A Survey and Assessment, by S. J. Petracek et al. of SRI (July 1976), prepared for the U.S. Dept. of Transportation/Transportation Systems Center under Contract DOT-TSC-968.

**In a large flat yard, the "top" half of the yard may be configured as in Figure A-1, with the "bottom" half a mirror image.

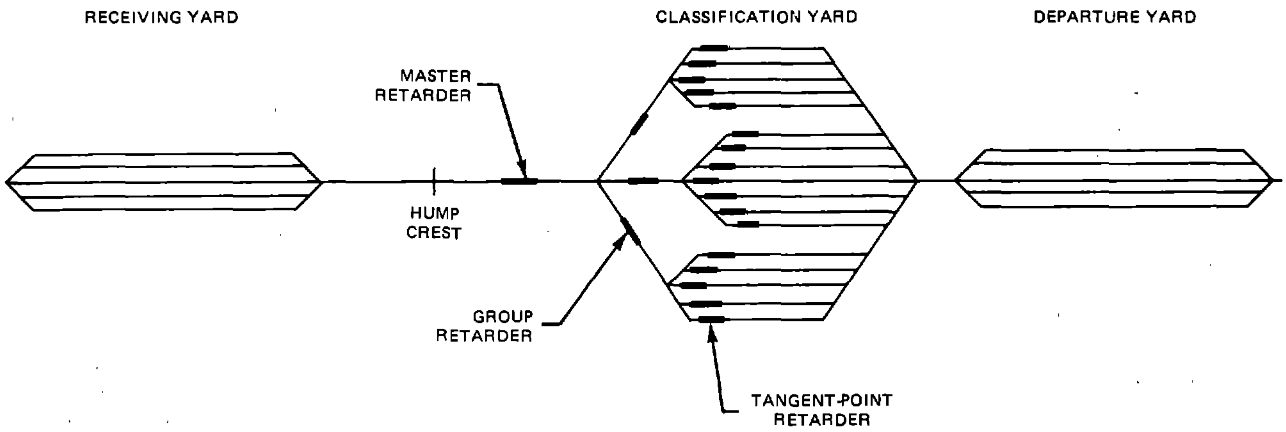


FIGURE A-2. SCHEMATIC REPRESENTATION OF A HUMP YARD

have a track available for yarding the inbound train, the train is held on the main line. If the incoming train is too long to be yarded on one track, it is broken up and yarded on two tracks. This "doubling" process requires between 10 and 20 minutes and can often be performed by the road-haul crew, depending on labor agreements and other considerations. After an incoming road-haul train has been yarded, the locomotive units and caboose are detached and moved to a service area. In some yards, however, the caboose is not detached but is actually humped (or switched) and sorted with the rest of the cars on the train.

After the incoming train or drag of cars has been yarded and the engines and caboose have been detached, the air-brake systems on the cars are ready for bleeding. This process is required because, after the engines are uncoupled (thereby breaking the air-hose connection), the individual car brakes are automatically activated by air in the compressed-air reservoir. Car bleeders must release this reservoir air to deactivate the brake system so that switch engines can freely push cars to the hump or along the switch lead. The air brakes are bled by carmen who walk along one side of the train and stop at each car to open angle cocks that release the air. Often, however, the release rod on the carman's side may be broken and he must climb or crawl between cars to use the release rod on the other side.

Generally, while the air-brake reservoir is being bled the individual cars are inspected for mechanical or physical defects. Some common defects are dragging equipment; mechanical failure of the air-brake system; cracked or broken wheels, bearings, and journals; broken couplers; door and seal problems; and car structural damage. Other defects, such as damaged or shifted loads, are also identified. After the "bad-order" cars are identified, they are sorted out from the others during the normal switching or classification process. Car identification and train consist information are also checked during this walk-by inspection.

The switching or classification process is the central activity in classification yards. It involves sorting the cars that have arrived grouped on a train or industrial drag into appropriately assigned classification (or "class") tracks. The assignment of cars to the classification tracks is generally based on the cars' destinations and the sorting policy of the yard. For example, in one yard, all cars bound for Chicago may be placed on Track 9 and all cars bound for Buffalo and Boston may be grouped together on Track 4. Other track assignments may be based on the condition of the car (such as whether it requires cleaning or repairs). Cars in transit through a yard often must be reswitched or rehumped for a number of reasons. For example, cars that require special processing (such as cleaning or repairing) usually must be reswitched after such processing has been completed. In addition, many yards do not have enough tracks to assign dedicated tracks to each of the blocks being made up in the yard. This forces yard personnel to mix blocks together on a track and then reswitch these cars when a slough track becomes available.

After being switched onto the correct classification track, the cars generally wait while others are being switched. The time that cars spend on a classification track waiting for enough other similarly bound cars to make a train is referred to as "accumulation time."

After enough cars have been accumulated to make a train, or according to a departure schedule, they are assembled into a train or industrial drag. In this process, the blocks of cars that are on a number of different tracks are joined together on a departure track that is long enough to hold all the cars for the train; lacking that, many yards use the main line for making up trains.

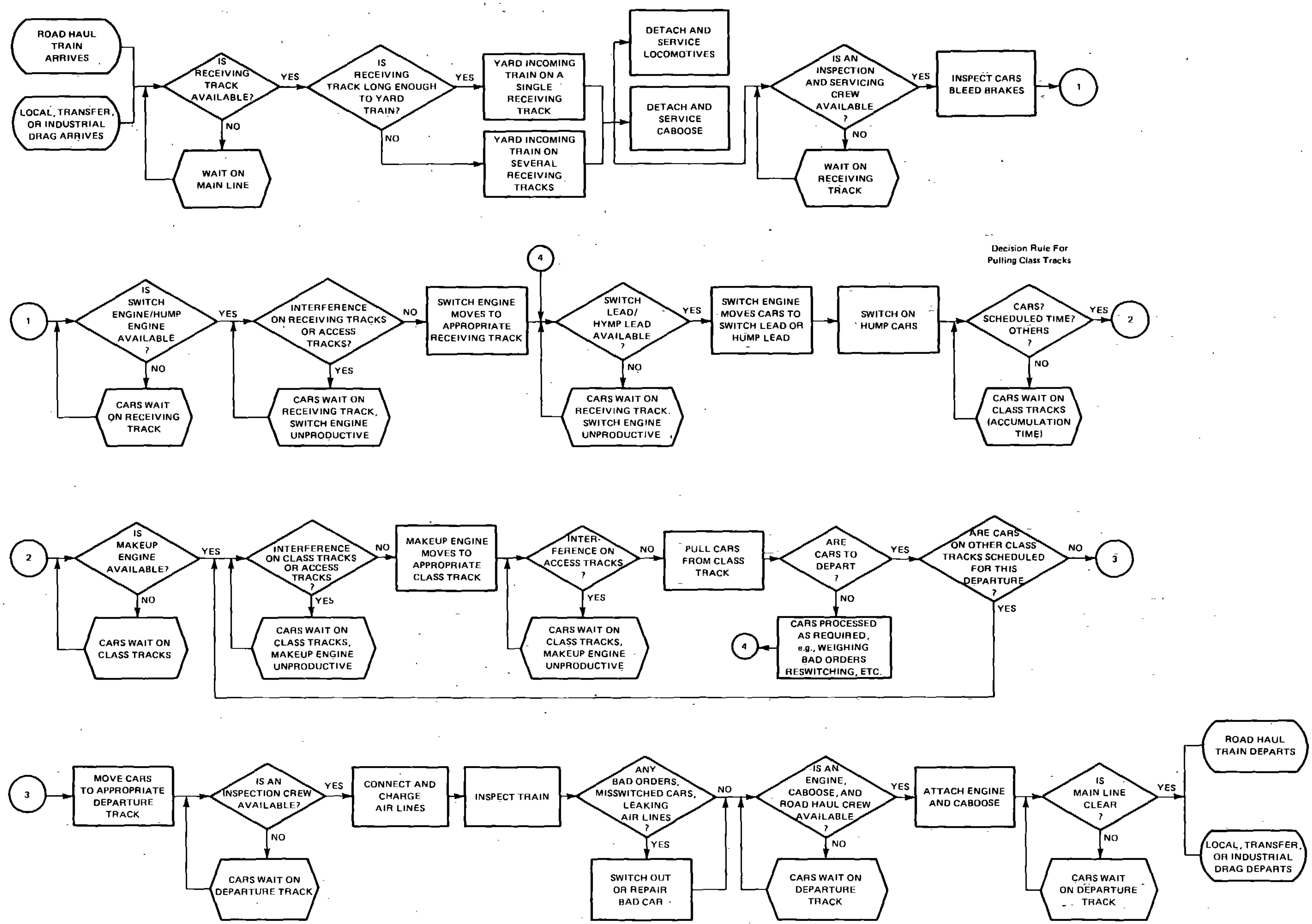


FIGURE A-3. CAR-HANDLING PROCESSES IN CLASSIFICATION YARDS

After the cars and blocks have been coupled, carmen connect the air-brake hoses, turn the brake valves, and inspect for bad-order or misswitched cars that must be switched out of the train. After the cars and air-brake hoses have been connected, the train's air-brake system is charged. There are three methods for charging the air lines. The first, and generally the fastest and most desirable, is to use sources of compressed air that are located near the make-up tracks. When such facilities are not available, the airing can be done by a switch engine or by the road-haul engines after they have been attached. The air-brake system is then checked for leaks.

At this time, the power units and caboose are attached and the train is physically ready to depart. A number of factors, however, can delay departure, such as lack of road-haul crew or power, lack of paperwork, or interference within the yard or on the main line.

This description of the major car-handling processes that take place in classification yards has been general in scope and only describes major classification yard operations. Although this description has been organized in a step-by-step operational sequence, it is important to note that, in most classification yards, these operations are performed simultaneously on different cars and trains. While one train is being received into a yard, another train or group of cars may be in the process of being classified, while still other cars are being assembled into an outbound train.

A.3 PROCESSING OF INFORMATION AND PAPERWORK

The actual movement of cars through a classification yard is usually paralleled by the processing of information and paperwork within the yard. In fact, the transference and processing of information is an essential supportive part of the car-classification and train-makeup processes. The purpose of these activities is to obtain a timely and accurate inventory of which cars are located on what tracks. Without such information, the classification and train-makeup processes could not be adequately controlled.

The first information a yard receives concerning an inbound train is its inbound consist, which is a description of the makeup of the train. At most large and medium-sized yards within modernized railroad networks, this consist information is received before the actual arrival of the train. Although this may allow the yardmaster some time for advance operational planning, this capability is often limited by the quality of the information or the amount of confidence the yardmaster has in it. The advance inbound consist is the outbound consist of the last yard where the train stopped, and that yard has the capability of revising the consist information. If the train picked up or set out any cars or blocks of cars after passing that terminal, the advance consist would be in error.

Additional information is obtained when the train arrives. The identification numbers of the cars in the train are noted and recorded using closed-circuit television and videotape or audiotape or by pencil and paper. These numbers can be checked against the advance consist inforation, and corrections can be made on an exception basis. Waybills and/or bills of lading also arrive with the train. A bill of lading is the agreement between the shipper and the railroad concerning the transportation of the shipper's goods. A waybill is a receipt that details the shipment and routing inventories. Locally originated traffic is often accompanied only by bills of lading, thereby requiring the preparation of waybills; inbound road-haul traffic is generally accompanied only by waybills. All this paperwork is the responsibility of the train conductor until the train reaches the yard, at which time a yard office clerk assumes responsibility. Waybills are then prepared from bills of lading for those cars that require them and, in some yards, punched cards are also prepared. The waybill and the other recorded information are then used to update the advance consist information. From this enhanced information, classification tracks are assigned and the switch lists are prepared. A switch list (also known as a cut list or hump list in hump yards) assigns a classification track for each car or cut of cars.

Thus, the actual switching of a train cannot begin until all this information processing has been accomplished. A potential for delay therefore exists if the information processing takes longer than the brake bleeding and car inspection. Slow rates of information processing may also limit the speed of processing high-priority trains and the effectiveness of certain automation features such as high-speed car-inspection systems or automatic coupler and brake systems.

After the switch lists have been completed, they are distributed to the yardmasters, hump foremen, retarder operators, switchmen, and the switch-engine crews. Any changes are then manually recorded on the switch lists. An example of such a change would be the reassignment of a car from a specific classification track to a bad-order track because of deficiencies discovered during the intial car inspection.

When the next step in the classification yard procedures is performed--the assembling of cars from different classification tracks into an outbound train--the waybills for the cars on the outbound train are also assembled. The car numbers and waybills are compared in an outbound check, and the necessary corrections to the consist are made. The waybills are then transferred to the outbound train conductor. When the train departs, the outbound consist is transmitted to a central processing location or to the next yard, where it becomes the advance inbound consist for the train.

Two major types of yard inventory systems are used to monitor car location and track status in the yard: manual and computer inventory systems.

An example of a manual system is the PICL system (perpetual-inventory and car-location system). In this system, punched cards are produced representing each car. A box (or rack) of pigeonholes is used to represent each track in the yard. A yard office clerk, referring to the switch list, manually sorts the waybills and punched cards into the pigeonholes corresponding to the tracks onto which the individual cars are switched. In this system, the waybills and punched cars follow the physical car movement from track to track by moving from pigeonhole to pigeonhole. Yard personnel can identify the cars on any track by looking at the punched cars in the corresponding pigeonhole. The computer inventory system is often referred to as a disc-PICL system (as opposed to the manual card-PICL system). The computer system keeps track of car location and status as cars are physically moved by appropriately processing computer files. Cars usually are moved in the computer according to the switch list or a prescribed classification track assignment table. The computer systems are much faster and more flexible and accurate than the manual systems; they normally can provide a track inventory, find the location of cars, and produce many types of management reports.

APPENDIX B: CURRENT COMPUTER INSTALLATIONS SURVEY

The use of digital computers in railroad yards is presented in the following tables:

- Table B.1--Yard Configurations
- Table B.2--Computer System Configurations
- Table B.3--Peripherals Used.

These tables are combinations of tables presented in "Computer Applications to Classification Yards," AREA Bulletin 650, pp. 272-277; and "Railroad Freight Car Classification Yards: Installations, 1978-1924," by WABCO.

Table B.1

YARD CONFIGURATIONS

	Gateway (MOPAC)	Eugene (SP)	Brosnan (SOU)	Bellevue (N&W)	Bailey (UP)	Perlman (P.C.)	No. Kansas City (BN)	Argentine (AT&SF)	Buckeye (P.C.)	Pasco (BN)	Roanoke (N&W)	Centennial (T&P)	Beograd (YRR)	E. Los Angeles (UP)	Calder (CN)	Alyth (CP)	Sheffield (SOU)	W. Colton (SP)	Inman (SOU)	Lang (D&TSL)	Northtown (BN)	Strawberry (L&N)	Rice (SCL)	Norris (SOU)	Barstow (AT&SF)	Englewood (SP)	Queensgate (CHESSIE)	Hinkle (UP)	WB N. Platte (UP)	DeWitt (CONRAIL)+
Single Lead-Single Hump	X	X	X				X	X	X	X		X		X	X	X	X	X	X	X	X			X	X	X	X	X	X	
Dual Lead-Single Hump				X																		X								
Dual Lead-Dual Hump					X	*					X		X																	
Receiving Yard Tracks		15	8	9	7	9	18	10	7	5	20	14	11	3	7	20	6	9	1		11	24	28	12						
Classification Yard Tracks		32	50	42	64	70	42	48	40	47	55	44	48	16	34	48	32	56	12	20	63	49	64	56	48					
Departure Yard Tracks		15	8	23	6	18	15	9	8	5	8	9	0	4	8	20	6	12	1		9	14	14	13						
Car Retarder Operator	X	X	X	X	X	X	X		X	X	X		X	X			X				X			X		X		X	X	
Master Retarder Control	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	
Group Retarder Control	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X		X	X	X	X		X	X	X	X	X	
Intermediate Retarder Control											X							X												
Tangent Retarder Control																		X	X						X		X			
Skate Retarder Control																	X	X			X	X	X		X		X			
Hump Speed Control										X							X	X			X		X		X		X			
Hump Signal Control					X		X	X			X	X			X	X		X		X	X		X		X		X			
Coupling Speed Measurement											X				X								X		X		X	X	X	
Auto Trim, Bi-dir. Track																X					X		X							
Pull Down Tracking																	X	X					X							
Auto Calibration Data Collection		X		X	X		X	X			X	X			X	X		X			X		X				X		X	
Computer Simulation																		X	X				X				X	X	X	

Sources: "Computer Applications to Classification Yards," AREA Bulletin 650, pp. 272-277 (1975).
"Railroad Freight Car Classification Yards: Installations, 1978-1924," Westinghouse Air Brake Company, Swissvale, PA (1978).

*Layout designed to provide for future dual hump.

†Manual control.

+Information not obtained yet.

Table B.2

COMPUTER SYSTEM CONFIGURATIONS

	Gateway (MOPAC)	Eugene (SP)	Brosnan (SOU)	Bellevue (N&W)	Bailey (UP)	Perlman (P.C.)	No. Kansas City (BN)	Argentine (AT&SF)	Buckeye (P.C.)	Pasco (BN)	Roanoke (N&W)	Centennial (T&P)	Beograd (YRR)	E. Los Angeles (UP)	Calder (CN)	Alyth (CP)	Sheffield (SOU)	W. Colton (SP)	Inman (SOU)	Lang (D&TSL)	Northtown (BN)	Strawberry (L&N)	Rice (SCL)	Norris (SOU)	Barstow (AT&SF)	Englewood (SP)	Queensgate (CHESSIE)	Hinkle (UP)	WB N. Platte (UP)	DeWitt (CONRAIL)+
Single Computer																			X					X		X				
Single Computer & Relay Backup		X	X	X	X		X			X	X	X	X	X	X					X										
Separate MIS Computer(s)																	X	X				X	X							
Distributed/Multiple PC Computers						X		X	X							X	X	X			X	X	X		X		X	X	X	
Combined PC/MIS Computer							X		X							X					X									
Multiple MIS Computers									X							X	X					X	X							
Computer Memory*: 12K			X										X	X					X					X						
16K		X		X				X							X		X													
24K					X		X			X	X	X					X			X								X		
32K																X	X	X			X									
48K																						X	X							
64K						X			X																		X	X	X	
Honeywell (H-115)		X		X																										
Honeywell (H-516)							X	X			X	X			X	X														
GE PAC 4000	X																													
GE PAC 4020						X			X	X											X									
IBM (S-1800)					X																					X				
Data General														X			X							X	X					
DEC (PDP 8 and 8I)			X										X																	
DEC PDP 11																			X	X							X†	X#	X†	
Xerox (Sigma 3)																		X												
Modular Computer Corp (Mod Comp II)																						X	X							

Sources: "Computer Applications to Classification Yards," AREA Bulletin 650, pp. 272-277 (1975). "Railroad Freight Car Classification Yards: Installations, 1978-1924," Westinghouse Air Brake Company, Swissvale, PA (1978).

---

*This refers to memory size for PC except at Sheffield and where PC and MIS are combined.

†PDP 11/60.

#PDP 11/34.

+ Information not obtained yet.

Table B.3

PERIPHERALS USED

	Gateway (MOPAC)	Eugene (SP)	Brosnan (SOU)	Bellevue (N&W)	Bailey (UP)	Perlman (P.C.)	No. Kansas City (BN)	Argentine (AT&SF)	Buckeye (P.C.)	Pasco (BN)	Roanoke (N&W)	Centennial (T&P)	Beograd (YRR)	E. Los Angeles (UP)	Calder (CN)	Alyth (CP)	Sheffield (SOU)	W. Colton (SP)	Inman (SOU)	Lang (D&TSL)	Northtown (BN)	Strawberry (L&N)	Rice (SCL)	Norris (SOU)	Barstow (AT&SF)	Englewood (SP)	Queensgate (CHESSIE)	Hinkle (UP)	WB N. Platte (UP)	DeWitt (CONRAIL)+
Disks		X			X		X	2		X	X	X				2	X	2			X	2	2				X	X	X	
Fixed Head Drum		X		X		X			X	X					X						X									
Cassettes																								X						
Line Printer					X	X			X	X						2	X				X						X			
Card Reader		X		X	X	X	X		X	X	X	X			X	2		X			X	1	1				X			
Card Punch		X		X	X	X	X		X	X	X	X			X	2		X			X	1	1							
Cathode Ray Tube (Output Only)							2	3	X	X		2					X	5			X	2	3				4	X	2	
Cathode Ray Tube (I/O)									X	X		3				7	X				X	3	2					X	4	
I/O Typewriters*		7	X	4	5	X	5	7	X	X	5	5	X	X	3	12	X	2	X	3	X	5	3	X			2	X	4	
Mag Tape																											X		X	
Paper Tape Reader/Punch																			X	X										
Note								†	†		†							†		†		†	†							

Sources: "Computer Applications to Classification Yards," AREA Bulletin 650, pp. 272-277 (1975).
"Railroad Freight Car Classification Yards: Installations, 1978-1924," Westinghouse Air Brake Company, Swissvale, PA (1978).

*Including "receiving only" TTYs.

†Does not include peripherals for MIS except for combined PC/MIS configuration.

+Information not obtained yet.

☆U.S. GOVERNMENT PRINTING OFFICE: 1981 522-585/14 1-3