LARGE SYSTEM OVERVIEW

INTRODUCTION

Categories Of Computers

- Microcomputers
- Minicomputers and Workstations
- Mainframes
- Supercomputers

Microcomputers

- Originated in late 70's
- Single-user-oriented
- Multi-user super micros
- PC, Apple Mac

Minicomputers and Workstations

- Smaller versions of mainframe
- Usually run on either the highly portable Unix OS or close OS such as VAX (DEC), Nighthawk, IBM AS/400, IBM RS/6000

Mainframes

- Oldest form of computer
- Expensive and powerful
- Small, Medium, and Large-Program handling
- Uniprocessor; high-speed storage
- Can be shared by a wide variety of users
- System/360, 370, 390 from IBM

Supercomputers

- Advanced technology
- Designed for advanced scientific calculations
- Most powerful and most expensive
- Equivalent to 50,000 PCs
- Cray 1, Cray 2, Cray X-MP, ETA-10.

EVOLUTION OF IBM MAINFRAME SYSTEMS

IBM Mainframe designs have compatible evolution based on System/360,and extended through System/370, System/370-XA, and now System/390. Prior to system/360, which was developed in 1964, there were different families of different systems for scientific computing and commercial data processing.

System/360
- Developed in 1964.
- DOS was the operating system
- Initial version of OS did not have the virtual storage concept

System/370
- Announced in 1970
- An upward compatible replacement for System/360
- Increased computing and I/O speed.
- Added interactive computing
- Models: 370/158, 370/155, and 370/165.
- Virtual storage concept introduced
- Use 24 bits of address total refer to main storage
- S/370-XA provides 31-bit addressing; that is, storage of 2 GB

System/390
- System/390; introduced in 1994 (latest version)
- CMOS technology
- MIPS
- Dual processing; fastest G4 model
- OS/390 – truly an enterprise server
- Incorporates MVS V5R2.
- Unix functions added.
- C and C++ compilers
- Java and Java Development kit
- GUI offered via Visual AGE Objects
- System can be easily moved

WHY MAINFRAMES?

- Can handle large volumes of data
- Have high processing power
- Provide centralized administration and storage
- Offer superior data management capabilities
- Can handle different types of workload
- Have high data bandwidth
- Monitor data integrity and security

Mainframe Operating Systems

MVS/XA, MVS/ESA, OS/390

- Suitable for production environment
- Transaction processing
- Time sharing
- Program development
- Virtual storage concept
- Multiprogramming, Spooling, time sharing and batch processing
- Addressing capacity
- Supports bimodal processing

VM/ESA

- Suitable for system software development
- Time sharing
- Electronic office
- University type of environment
- Supports multiple virtual machines
- Can host other OS's
- Uses virtual storage

OS/390

Prior to OS/390, the System/390 world consisted of tons of different products that had to be ordered, installed and maintained separately. The products were updated in different release cycles.

OS/390 An Integrated Solution

The OS/390 system builds on the strengths of MVS/ESA and provides a complete network-ready server environment.

Large Systems Components

Hardware

- Central Electronic Complex
- Central processors
- System controller
- Central storage
- Expanded storage
- Channel subsystem

INPUT/OUTPUT

- Control units
- Devices: DASD, tape, printer

Communication Subsystem

- Communication controllers
- Cluster controllers
- Terminals
- Multiplexes, modems

SOFTWARE

- Operating system (like MVS/ESA)
- System applications
- Job entry subsystem
- Communication software
- Batch applications
- DB/DC Applications
- Storage management subsystem

Categories of Programming

- Batch programming
- Interactive programming

Batch Programming

- Accepts data for processing in groups called batches and produces o/p grouped into sets.
- Do not communicate with any user as they execute
- Once started, supervised by the computer's OS

Interactive programming

- Designed to communicate with users as they run
- Issues messages and prompts
- Allow the end user to communicate new info or change existing info
- Adapted with CICS

Chapter 2

Hardware overview

Central Electronic Complex

Central Processors

- Enterprises systems,
- Processor types 9221
  9121
  9021 (fastest)
- Uniprocessor - One CPU
- Dual processor - Two CPUs
- Dyadic processor - Two CPUs
- Triadic processor - Three CPUs
- Tightly coupled - One copy of the operating system
- Loosely coupled - Two independent systems connected together

System Controller

Monitors communication among central processor, other processors, storage and channel subsystem

Central Storage

- Main memory, also known as RAM
- Loaded with system programs, user programs, and data

Expanded Storage

- Optional high-speed memory
- Slower than main memory
- Less expensive
- Used mainly as an I/O device (for paging)

Input/Output subsystem

- Channel subsystem
- I/O processor
- A set of RISC based processor from I/O handling

Control Unit

- Controls I/O devices
- Device diagnostics
- Selects device

Addressing Modes (AMODE)

- IBM supports two types of Addressing Modes:
- 24 bit addresses – prior to XA – Address space size 16 MB
- 31 bit addresses – XA and after – Address space size 2 GB

Logical Partitioning (LPAR)

With PR/SM (processor Resource/ System Manager), a single system can be made to support many operating systems. The logical Portioning offers multiple environments. A single system's resources are portioned. Different OS's can run in these partitions.

Communications subsystem

- Communications controller
- Provides networking
- Remote device connectivity
- Controls network communication devices

Modems

Changes digital signals to analog signals and vice versa

Chapter 3

The Operating System

Operating System is a maser system program that controls a computer system. When there are a number of jobs to be run in a batch processing, instructions are written to OS among other things,

- Performs system scheduling tasks
- Handles system interruptions
- Monitors system status

Operating system characteristic

- Starts/stops applications
- Multiple users
- Multiple terminals
- Multiple devices management
- Network interfaces management
- Prioritizes work
- Job processing and management
- Evolutionary
- Grow able
- Reliability/Availability/Serviceability
- Security
- Re-configurable
- Measurable/Tunable

MVS Evolution

- 1961 OS 360 PCP
- 1966 OS/MFT

OS/MVT

- 1972 OS/VS1, OS/VS2 R1 (SVS)
- 1974 OS/VS2 R2 (MVS)
- 1983 MVS/XA
- 1988 MVS/ESA
- 1995 MVS/ESA 5.2.2
- 1996 OS/390 1.1
- 1998 OS/390 2.6

Function of OS

The Functions of OS are

- System Management
- Resources Management
- Memory Management
- File Management

System Management

- Starts and shut down the system
- Manages network interfaces
- Prioritizes the work in TSO mode
- Job processing
- Security management

Resources Management

- Decides which devices to be accessed by the CPU
- Prioritizes the work for the devices
- Manages the CPU for the optimum usage
- Establishes the connectivity.
- Loads compilers, applications
- Manages the library routines

Memory management

- Allocation of memory
- De allocation of memory
- Fragmentation and garbage collection
- Paging and segmentation.

File Management

- CREATE
- DELETE
- COPY
- MERGE
- VIEW
- ADD
- EDIT
- PRINT
- OPEN
- CLOSE

Characteristic of an OS

- Evolutionary
- Re-configurability
- Reliability
- Availability
- Serviceability
- Security
- Measurability

Characteristic features of Mainframe OS

The distinction is slowly disappearing among the various types of computers.
But the level of sophistication is very high in mainframe machines.

The five characteristics are
Virtual storage
Multi-programming
Spooling
Batch processing
Time sharing

Virtual Storage

The processors main storage is among the most valuable of the system resources.

Modern mainframe computer operating systems provided sophisticated services to make the best use of the available main storage

Most important of these services is virtual storage

Virtual storage is a technique that lets a processor simulate an amount of main storage that is larger than the actual amount of real storage.

To do this, the computer uses disk storage as an extension of real storage.

In virtual storage, at any given moment, only the current program instruction and the data it access need o be real storage.

Virtual storage is largely transparent from the user's point of view.

Virtual storage appears to be real storage to the user.

Multiprogramming

Multiprogramming simply means more than one program executing at the same time.

Actually that is misleading. At any given moment, Only one program can have control of the CPU.

A multiprogramming system appears to execute more than one program can have control of the CPU.

A multiprogramming system appears to execute more than one program at the same time.

EXAMPLE Reading data from an input device takes more than any other Operation.

As a result most programs have to wait for a large portion of time for the I/O Operations to complete.

If programs were run one at a time on a mainframe, the CPU would spend most of its time waiting.

Multi Programming like virtual storage is mostly transparent to the user.

Each user thinks that he has exclusive right to use the system.

Spooling

A significant problem that must be overcome by multiprogramming systems is sharing access to input and output devices for the programs that execute together.

For Example spooling is used to provide shared access to printer devices.

Benefit of spooling is used to provide shared access to printer devices.
The expensive processor has to wait for I/O
The speed mismatch is of the order of 1:1000.
To reduce the delay a low cost I/O processor is used to prepare input data and print results, thereby main computer dose not wait for want of I/O.

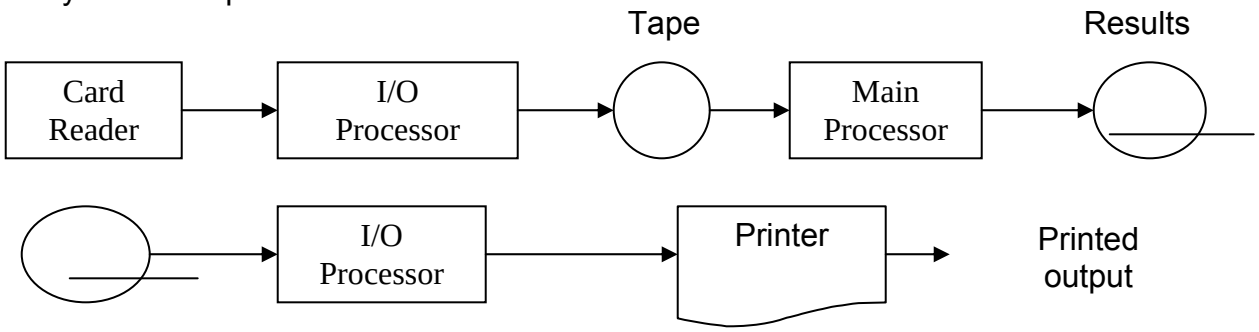


Figure 1-1

The reason for using an I/O processor was to keep programs and data ready for main processor to process.

The programs are transferred in batches to high-speed tape. Each program will have begin and end card.

The method of putting inputs into a batch and placing them in a magnetic medium and queuing the output is called spooling.

Advantages

The computers could be so scheduled that while present batch is processed the previous batch results could be printed out and data for next batch could be prepared.
Useful in engineering/scientific computations where processing time is large while I/O is not that large.
Time wasted between manuals loading of successive jobs is reduced.
The I/O processor and main processor work independently & simultaneously, thereby increasing the total number of jobs to be processed.
The main computer I/O is via magnetic tape or disk, which is faster than keyboard or line printer. Thus, memory and processor do not waste time for I/O.

MVS Components (MVS/ESA)

SP – System Product.
BCP – Basic Control Program
JES – Job Entry System.
DFP – Data Facility Product
BCP manages work during execution.
JES manages work before and after execution.

MVS/ESA supports
Multiprogramming
Time sharing
Multitasking
Multiprocessing
TCMP – Tightly Coupled Multiprocessing
LCMP – Loosely Coupled Multiprocessing

Virtual Storage

The basic approach is to divide total programs into small sequences of instructions called either pages or segments. Then only those program pages or segments that are actually required at a particular moment in the processing need be in real storage from where they can be rapidly retrieved as needed. The OS handles the swapping. Thus, form the application's point of view; the effective (or virtual) size of the available primary storage may appear to be unlimited.

Virtual Storage – Mapping

Virtual storage is divided into pages of 4K sizes.
Real Storage is divided into frames of 4K sizes.
Real storage can be central storage or expanded storage.
Virtual storage is what the programmer “sees”.
Real storage is what the system has.
The mapping of virtual storage into physical storage is accomplished by the OS, which requires that the virtual addresses in a program be translated into real address.
Dynamic address translation (DAT) is a hardware feature, which translates virtual addresses into real addresses when instructions are executed.
An address space can range from 0 to 2 GB.

The address space is divided into two areas.

Common area or the system area that contains the operating system and other common programs used by all the users and private area, which contains the user program.

The common area is mapped around the 16MB line in the address space o provide the bimodal operation of supporting both 24-bit programs and 31-bit programs.

BCP – Basic Control Program Functions

Processor Management
I/O Management
Work Management
Multiprogramming and swapping
Memory management – Real storage manager (RSM)
Auxiliary storage manager (ASM)
Virtual storage manager (VSM)
Resource management – System resource manager (SRM)

System Resource Manager

SRM manages the resources of the system in such a way that the goals of the installation are met. The goals can be in terms of:

Throughput – The total amount of useful work done by the system.
Response time – Time taken to receive response for a command.
Turnaround time – Time taken to get the output after submitting on job.

Paging

Pages in virtual storages are mapped onto frames in real storage or slots of the auxiliary storage (DASD) or both. Pages are brought to central storage when they are required, which is called page in. Inactive pages from central storage are removed to auxiliary storage which is called page out.

Data in memory

Accessing data from a dataset in auxiliary storage requires I/O operations and will take time. Bringing data into memory, namely virtual storage and accessing the data from there will minimize I/O and the elapsed time for an application.

MVS/ESA provides data spaces, which are also virtual storage objects where large volumes of data can be stored and retrieved faster.

Job Entry Subsystem (JES):

JES manages the jobs that enter the system. IBM supports two products namely JES2 and JES3.

Batch Processing

A batch file contains a series of commands that are processed together as a batch.

While using batch processing, work is processed in units called Jobs.
A job may cause one or more programs to execute in sequence.
Jobs are submitted in batches such that they are processed one at a time.
Many users compete to use the system's resources.
To manage this, the job entry subsystem, or JES processes each user's job in an orderly fashion.

Job Classes

An installation can define job classes of its own based on different criteria like time, resources required for the jobs. Jobs can be submitted in any of these defined classes.

When JES starts, it defines automatically a number of batch or initiator address spaces.
Initiators can be associated with certain classes of jobs.
For ex. Initiator 1 with class A, Initiator 2 with class B, and Initiator 3 with class X.

The users submit batch jobs and they are run in batch address spaces or initiator address spaces. Job can be submitted from a local terminal, remote terminal or from a different system. These are called lje, rje, and nje respectively.

Steps In Job Processing

Jobs are read by JES into the spool.
Jobs are then checked for syntax errors and converted into some internal form.
A job is selected and put into an initiator address space.
The job is executed.
The outputs are cauterized according to output classes.
The outputs are printed out.
Job is removed from the system spool area (Purging)

Job Control Language (JCL)

Jobs are defined using the job control language. JCL contains three important statements: JOB, EXEC and DD statements.

JOB STATEMENT

Introduces a job to the system.
Identifiers class of work
Provides accounting parameter
Provides scheduling information

EXEC STATEMENT
Specifies what is to be done or which program to run.

DD STATEMENT
Allocates/Catalogs datasets.
Links programs and datasets
Concatenates data sets
Controls output destination.

Procedures
JCL statements can be combined and stored as procedure.

Time-Sharing

In a time-sharing system, each user has access to the system through a terminal device.

Time-sharing is sometimes called interactive processing, because it lets users interact directly with the computer.

Sometimes, time-sharing processing is also called foreground processing, while batch job processing is called background processing.

Types Of Programs

I/O Bound Programs

In programs when large quantity of data is required to be read, or printed, the CPU has to wait for large portion of time waiting for I/O. Such programs are called I/O bound programs.

Example : Commercial Applications.

Processor Bound Programs

In engineering & scientific applications, I/O/ is less while processing requirements are more such programs are processor bound programs.

Example : Missile Tracking, Weather forecasting, remote sensing

Operating Systems of the DOS family

DOS means Disk Operating System, which was originally designed for small system configurations that had limited processing requirements.
DOS is commonly called DOS/VSE.

VSE stands for virtual storage extended, which refers o, the particular way it handles virtual storage.

Although DOS/VSE has evolved into a respectable operating system, it is still appropriate for smaller systems that do not have extensive processing requirements. For larger configurations, the OS family of operating systems is more appropriate.

The OS Family of Operating System

Designing for installations that required the full range of processing possibilities
Two versions of OS called OS/MFT and OS/MVT were in use.
They different in the way they handled multiprogramming.
The current form of OS is called MVS, which stands for Multiple Virtual Storage.
In MVS each multi-programmed job is given its own virtual storage address space.

The VM Operating System

A third IBM operating system is called virtual machine (VM), which has different approach to computer system management than DOS OS.

VM, which stands for virtual machine, uses a variety of techniques including virtual storage and multi-programming to simulate more than one computer system (called a virtual machine) on a single real computer system.

The operating system supports running other operating systems as its guests.

The host VM handles resource allocation to the various guests, which are running in different “regions” or partitions.

VM thus provides many different virtual machines with a single system.

VM provides a special operating system, called CMS for conversational monitor system, that lets a single terminal user use a virtual machine interactively.

An Introduction to MVS and TSO

This topic introduces the basic characteristics of MVS and TSO. One has to first understand the basics of its underlying operating system MVS before understanding TSO.

An Introduction to MVS

MVS, which stands for Multiple Virtual Storage supports TSO.

Although TSO stands for time-sharing option, it is not an option; it is an integral part of MVS. To use TSO effectively, one must understand the capabilities of MVS. MVS is available in three versions: MVS/370 MVS/XA and MVS/ESA.

Multiple Virtual Storage

The complete range of address is called the address space.
The number of digits used to represent addresses limits the maximum size of an address space.
Real computers use binary numbers rather than decimal numbers to represent addresses.
MVS uses multiple address space capability to Implement multiprogramming.
Each batch job running under MVS executes in its own address space.

Jobs and Job Control Language:

A Job is the execution of one or more related programs in sequence. Each program executed by a job is called a job step.

E.G. One wants to process a job that executes two programs: the first sorts a customer file into customer name sequence, and second prints a report that lists customers by name. This is a two-step job, because it requires two programs. No matter how many job steps your job steps your job contains, when you submit it to be processed by MVS, MVS treats it as a whole. In other words, every job begins with the execution of the first program and continues until the last program finishes executing unless an error occurs.

Within a job, the job control language, or JCL provides the specifications necessary for MVS to process the job. Although you do not have to know JCL to use TSO, you won’t last long in an MVS without JCL.

The Job Entry Subsystem (JES)

To process a job on an MVS system, MVS uses an important component called the job entry subsystem, or JES. Simply put JES is the MVS component that keeps track of jobs that enter the system, determine when they are executed, and sends each job's printed output to the correct printer.

IBM integrated the functions performed by these other programs into operating system by providing the job entry subsystem. There are two versions of JES, called JES2 and JES3.

JES2 is designed for single processor systems that do not have serious job scheduling problems. In contrast, JES3 is designed for large, multi-processor systems, where job scheduling is highly complicated. Each MVS system uses either JES2 or JES3; the two job entry subsystems cannot be used together on the same system.

When you submit a job for processing under JES2 or JES3, the job's JCL statements and other inputs are held in a job queue until JES2/JES3 determines that the job is ready to be executed. Then JES2/JES3 presents the job to MVS for execution. As the job executes, its printed output is collected in the spooled output on an appropriate printer.

With either JES2 or JES3, you can assign each job class and a priority to prioritize the jobs in the job queue. Then, JES2/JES3 executes jobs with higher class and priority assignments, even if jobs with lower class and priority were waiting in the queue longer. JES3 provides additional job scheduling functions that are not available under JES2.

JES functions

JES receives the job, checks for correctness of the JCL statement, translates JCL into internal form and places the job in a proper processing category. After the job is processed, JES arranges the job output to be routed to the destinations and finally removes the job related information from its spool space.

JES stages

Input - Job is read by JES into the spool
Conversion - JES is converted into internal form
Allocation - The requested resources are allocated to the job
Execution - The program is executed
Output - The program's outputs are classified
Hardcopy - The outputs are printed out
Purge - All the items related to the job are deleted from spool

Security

Standard MVS security

All IBM products have their own internal security mechanisms, but IBM recommends its products called RACF for security purposes.

RACF (Resource Access Control Facility)

It is responsible for all security in the system. Users can be authorized and resources can be authorized and RACF ensures that only authorized users can use the system and that too the resource authorized for them.

The RACF administrator crates user profiles, which can store the userid, password and other security information. RACF can also have resource Profiles, which allow specific user or groups of users, access these resources.

RACF can protect DASD volumes, tape volumes, libraries, transactions, logons, applications, terminals and any other resources.

Chapter 4

DATA Management

Dataset

A dataset is a collection of data in the form of records. In other words, a file is called a dataset in IBM terminology.

There are four types of Non-VSAM dataset
Sequential Dataset
Indexed Dataset
Direct Dataset
Partitioned Dataset

Sequential Dataset
This is nothing but the sequential file we are familiar with.
Records are added only at the end of the dataset
Records cannot be deleted
Records can be inserted at the end.
Supports fixed length or variable length records.
Used for sequential processing.

Indexed Dataset

Used or direct or random processing of data.
Application typically wants to access a particular record randomly.
A field in the generated and maintained, which is used to access the record using the key.

Direct Dataset

Used for direct or random processing of data.
Key field is used to access data.
Key value is converted to the record address by some hashing algorithm.

Partitioned Dataset
A collection of sequential dataset called members.
Has a directory to keep track of members and their locations.
Used for realizing a library structure.
Can maintain libraries of source program, load modules.

MVS/ESA Access Methods:

Non-VSAM access Methods:

Sequential	-	SAM, BSAM, QSAM
Indexed	-	ISAM, BISAM, and QISAM.
Direct	-	DAM, BDAM
Partitioned	-	BPAM + SAM

Four Types of VSAM (Virtual Storage Access Method):

Sequential	-	ESDS (Entry Sequenced Dataset)
Indexed	-	KSDS (Key Sequenced Dataset)
Direct	-	RRDS (Relative Record Dataset)
Linear	-	LDS (Linear Dataset)

VSAM Advantages:

High Performances indexed and direct datasets
Alternate Index support
Similar block management for all file types.
MVS uses VSAM intensively Example Catalogs are VSAM dataset
Path and upgrade service support for alternate index

TAM (Telecommunication Access Method)

VTAM - Virtual Telecommunication Access Method
BTAM - Basic Telecommunication Method
TCAM - Tele Communication Method

Access Methods acts as interface between application programs and the operating system or the channel subsystem.

The application programs raise I/O requests. They are handled by the channel subsystem independently, using the channel programs generated by the access methods.

DASD Datasets

Every DASD volumes have a VTOC.(Volume Table of content)
VTOC has entries for all the datasets residing in that volume.
VTOC entries typically contain names of the datasets, their attributes and their locations (addresses) in their volume.
VTOC also has information about free space in the volume.

Catalogs

Catalog is a dataset itself, which has entries to other datasets containing information as to which unit and which volume contains the dataset.

Catalog is used by the MVS to locate a dataset without the user having to give the location information. All datasets should be catalogued, so that he user need not remember where his datasets are located.

OS Catalogs

This was the first type of catalog introduced by IBM, which contained dataset names and their exact locations.

VSAM Catalogs

VSAM catalogs to store a lot more information about VSAM dataset.

Integrated Catalog Facility (ICF)

Later the above two type were integrated into the ICF structure.

The ICF structure contains two main structures:
The catalog itself called BCS (Basic Catalog Structure)
VVDS (VSAM volume Dataset)
The BCS contains all the dataset names, VSAM or Mon VSAM.
VVDS contains only VSAM dataset names and information about them.
VTOC always contains information about all datasets.

VTOC	One Per volume
VVDS	One Per volume
BCS	Any number of catalogs in the system.

Master and User Catalogs

Every system will have one and only one Master Catalog and any number of user catalogs.

Master Catalog contains entries for user datasets.
Important system datasets
Other user catalogs
Alias for user catalogs

User Catalogs contain entries for user datasets.

The User dataset is cataloged automatically in any particular user catalog depending on the user ID, which is always the first part of any dataset name.

Chapter 5

Batch Processing

Job, Program and task

A job can be considered as consisting of one or more job steps.
A job step is nothing but execution of a program.
A program is an executable called a load module
A job step internally becomes an MVS task

A program written in a high-level language or assembly language is called a source module.
A compiler or assembler into an object module translates a source module. into a object module.
Object modules are linked together by the linkage editor to create a load module, which is in a loadable and executable form.

The users submit batch jobs and they are run in batch address spaces or Initiator address spaces. Jobs can be submitted from a local terminal, remote terminal or from a different system. These are called LJE, RJE and NJE respectively.

Steps in a job Processing

Jobs are read by JES into spool.
Jobs are then checked for syntax errors and converted into some internal form.
A job is selected and put into an initiator address space.
The job is executed.
The outputs are categorized according to output classes.
The outputs are printed out.
Job is removed from the system spool area.

Job Control Language

Introduces a job to the system.
Identifies class of work.
Provides accounting parameter.
Provides scheduling information.

EXEC Statement

Specifies what is to be done or which program to run provides dispatching priority Information.

DD Statements

Allocates/ catalogs files
Links programs and datasets.
Concatenates data sets.
Controls output destination.

Procedure

JCL statements can be combined and stored as procedures.

Job Classes

An installation can define job classes of its own based on different criteria like time, resource required for the jobs, jobs can be submitted in any of these defined classes.

When JES starts, it defines automatically a number of batch or initiator address spaces. Initiators can be associated with certain classes of jobs. For ex, Initiator 1 with class A, Initiator 2 with class B, initiator 3 with class X.

JES maintains the job queue for different classes of jobs and selects the job from a class according to its selection priority or scheduling priority.

Temporary Datasets

Sometimes, an application requires to create temporary data sets and after use , these temporary work files can be deleted. They need not kept permanently nor they need to be catalogued. For Example, a sort program requires lot of work area for temporary storage, which can be deleted after the sort job is over.

In JCL, one can create a temporary data set in one step, use it in another step, but at the end of its use, system will automatically delete it, because it is a temporary data set.

Dummy Datasets

In certain situations, while testing the logic of a program, one may create a dummy file. This is called a dummy data set. A dummy data set is never created really. When one allocates a dummy dataset, and the program tries to read it, an EOF condition is encountered. When a program writes into it, nothing is written.

Dataset concatenation

Sometimes, the program wants to process a number of files one after the other, for the program, logically, it is a single file, but it may be consisting of physically different data sets. In this situation, one can “ concatenate” the different data sets by DD statements as if it is a single file.

Data set reassignment

DD statements are very useful for making programs device-independent or data set independent. Source programs do not contain reference to a physical data set name directly. Instead a logical file name or DD name is coded in the program. The JCL for executing this program will contain a DD statement, which refers to the actual data set name. This is very useful, because if you want to run the program with a different data set, it is sufficient to modify the DD statement in JCL, rather than modifying the program. (Modifying a program requires a recompilation and relinking, remember).

Chapter 6

Interactive Processing

Categories

Time Sharing or simple interactive processing

TSO/E
ISPF/PDF

Transaction Processing

One address space per user
User access to all (authorized) resources.
File/data base

TSO

Part of MVS/ESA
Command Language
Line Editor
Interactive Compile, Link, Debug
Batch job submission, retrieval

ISPF-Interactive System Product Facility

ISPF is a dialog manager, Provides dialog services. REXX or CLIST can be used to create dialogs.

CICS/ESA

DB/DC system
Transaction manager
Terminal/ Transaction oriented system
Multi-tasking
Single Address space architecture
Multiple Programme environments
Controlled user access

CICS/ESA Application Development

COBOL, PL1, Assembler, C

Command level Coding

CICS programming was done initially using macro level coding.
High-level CICS programming uses CICS commands embedded in programs is called command level coding, COBOL, C, RPG, PL1, Assembler languages can be used to write CICS programs.

CICS ISC (Intersystem communication)

CICS is one system talking to CICS in another system

CICS MRO (Multi Region Operation)

Normally CICS is single address space architecture. But it is possible to have more than one CICS address space in a single system.

The reasons for multiple regions can be:

One region for production and one region for testing
One region for one application and other for a different application
One region doing database/ file access and another for terminal control
One CICS address space talking to another CICS address space in the same system is called MRO.

Information Management system (IMS)

DB/DC system
Contains IMS/DB and IMS/DC components
Multiple address space architecture
Controlled user access
Supports Hierarchical model of database design
IMS address spaces

IMS control region

Data Language I region (DLI)
Contains resource lock manager (IRLM)
Message Processing Program regions (MPP)
Interactive Fast Path (IFP)
Batch Message Processing Region (BMP)
Batch Processing Program Regions (BPP)

IMS/ESA application development

Application Development facility II (ADF II)
Cobol, C, Pl/1, Pascal, Assembler
IMS Inter system communication (ISC)
IMS multi system Coupling (MSC)
IMS Extended Recovery Facility (XRF)

Cross system Product

Application Generator
Panel Driven
DL1 and DB2 support
Interactive Test facility
Easy to learn and use
High Productivity
Useful for quick prototyping
High-level Application portability among different platforms
Like a $GL
Like a CASE tool.

Chapter 7

Database

Large Volumes of data, which is independent or interrelated, have to be handled by the enterprise. Different users may view the same data differently and many applications of different type like batch and interactive applications should have access to the data.

Database is organized collection of
Independent data.
Interrelated data.
Minimal/controlled redundancy
Used by one/multiple application
Shared by many user/applications

DBMS Database Management System Objectives

Large Data Volume
Data redundancy reduction
Shared/dispersed data
Data independence
Interactive and batch processing
Simplified programming
End-User programming
Integrity
Security

Flat file approach

Entity relationship determined by program
There is a tight relation between the program and the structure of data.
Modifications to the data structure entails program modifications and vice versa.

Models of Database

Hierarchical	-	IMS
Network	-	IDMS
Relational	-	DB2

Choosing the database Model for your user:

Choose Relational if you have dynamic requirements Choose hierarchical if you have static

requirements□□Frequent change □Managed change□□Unplanned□Carefully planned□□

Advantages:

Hierarchical Advantages□Relational Advantages□□Good when resource consumption is more important than flexibility□Concepts easier to understand□□Some real life data has natural hierarchy□Simpler program/Tabular design□□Good performance for large stable database□Simplified database/Program maintenance□□

DB2 Database

Relational data model – Tables

SQL - Structured Query Language
DDL - Data Definition Language
DML - Data Manipulation Language

Data is stored in tales, identified by table name, column and rows

Views

View is made up from one or more tables.
Views do not physically exist in the database, except as a definition.
Views are “logical table” made up in “memory” when required.
Views are useful for security also

DB2 Database Processing

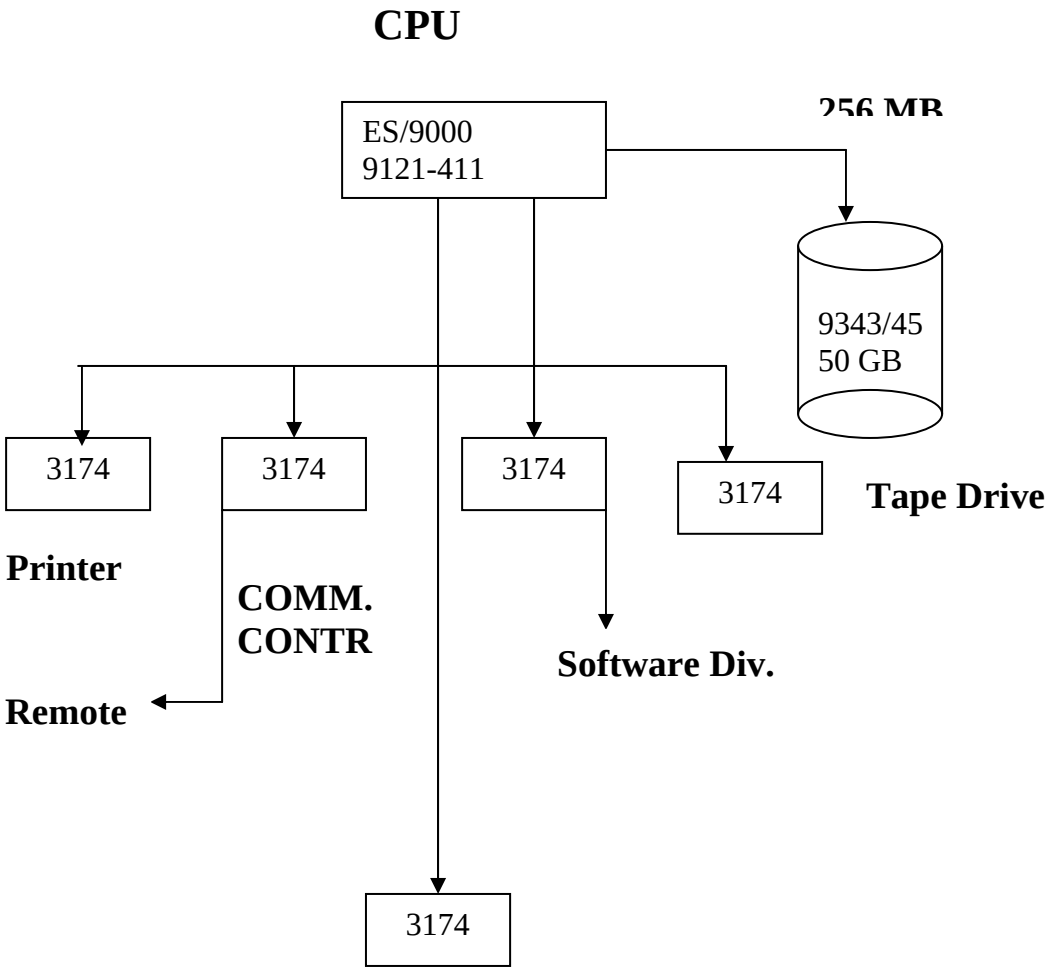
SQL statements

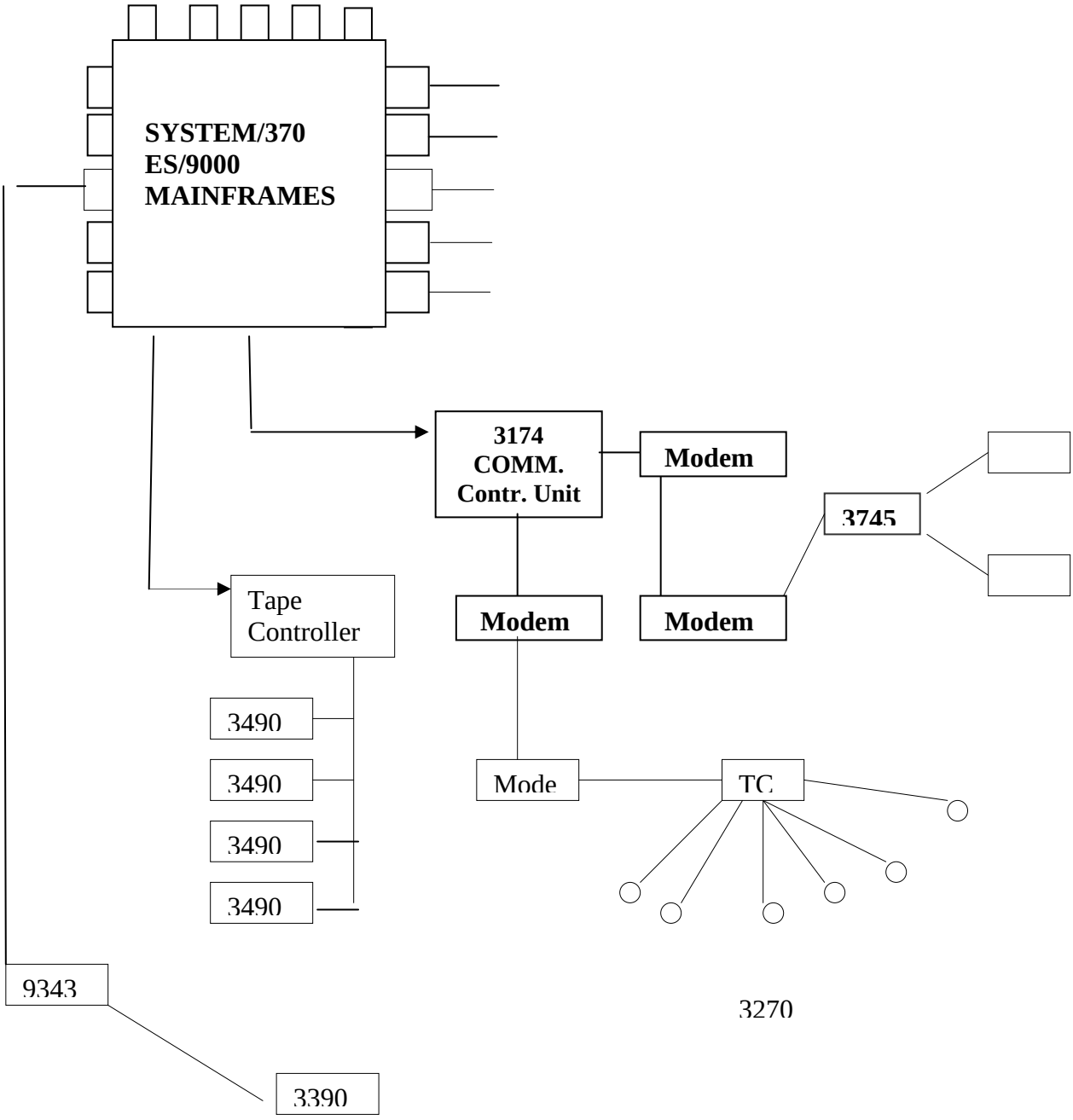
Different ways to access DB2 using SQL:
Embedded SQL in an application program
SPUFI
QMF

DB2 Interactive Interface

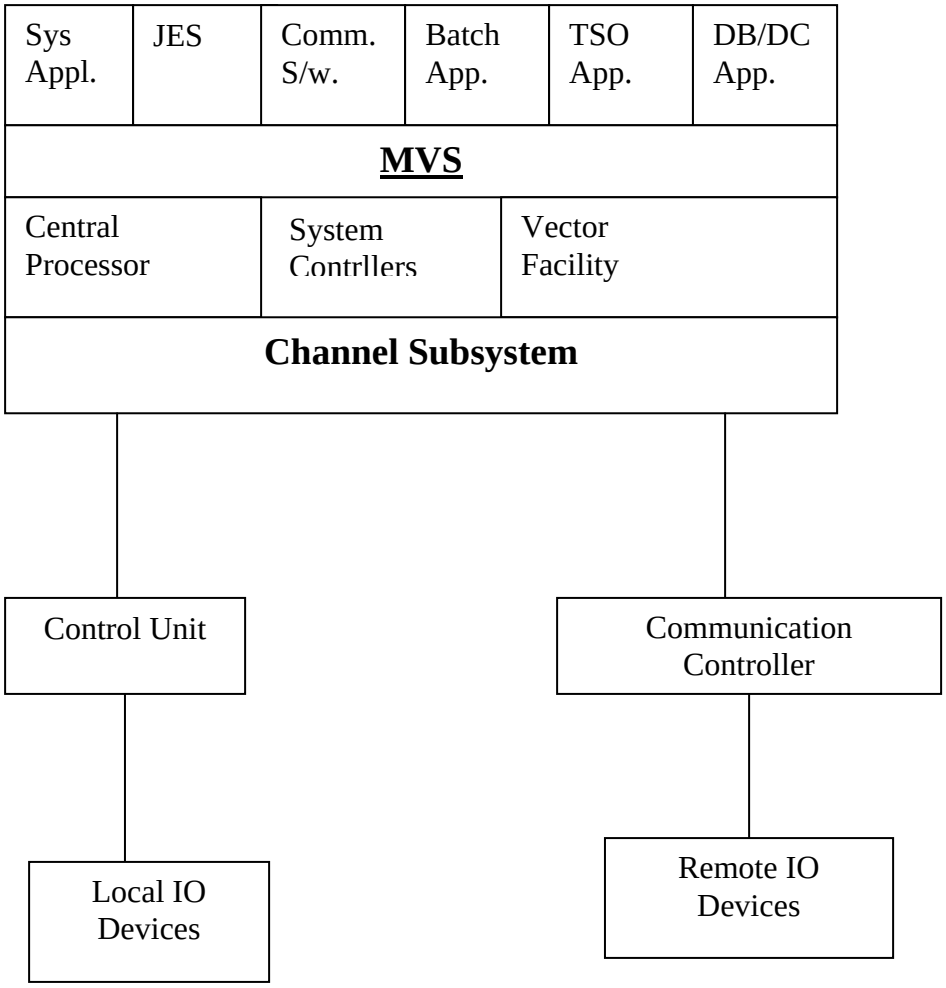
Supports SPUFI, DCLGEN, BIND, Program Preparation, RUN, DB2 commands, DB2 utilities etc.

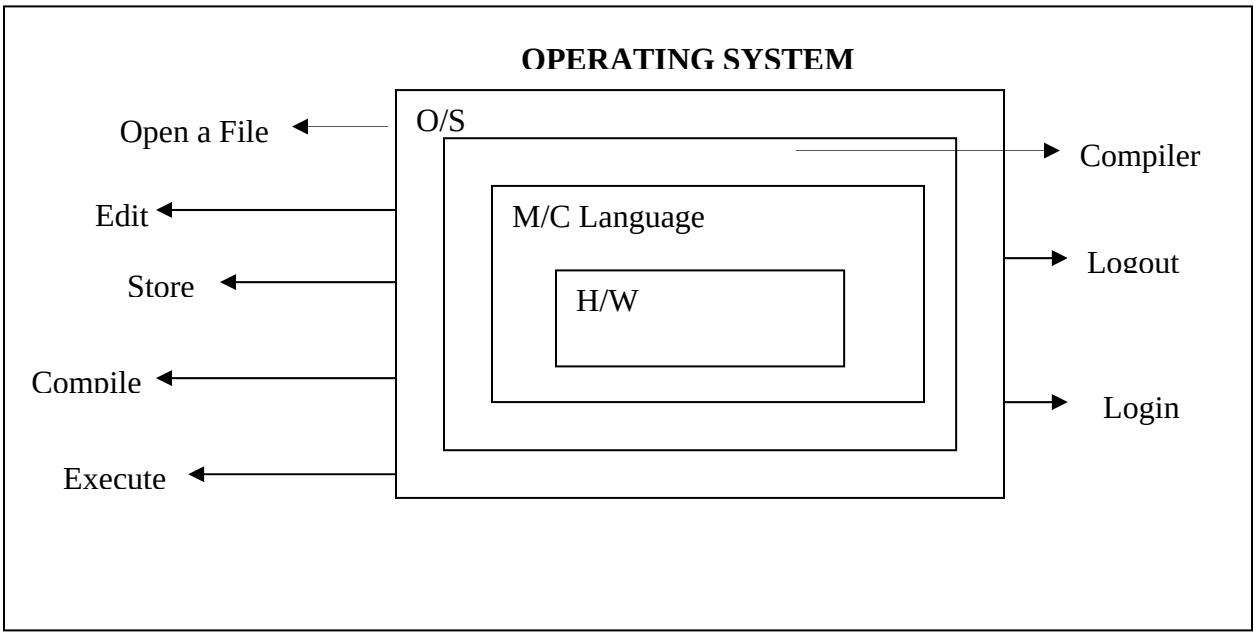
System Overview

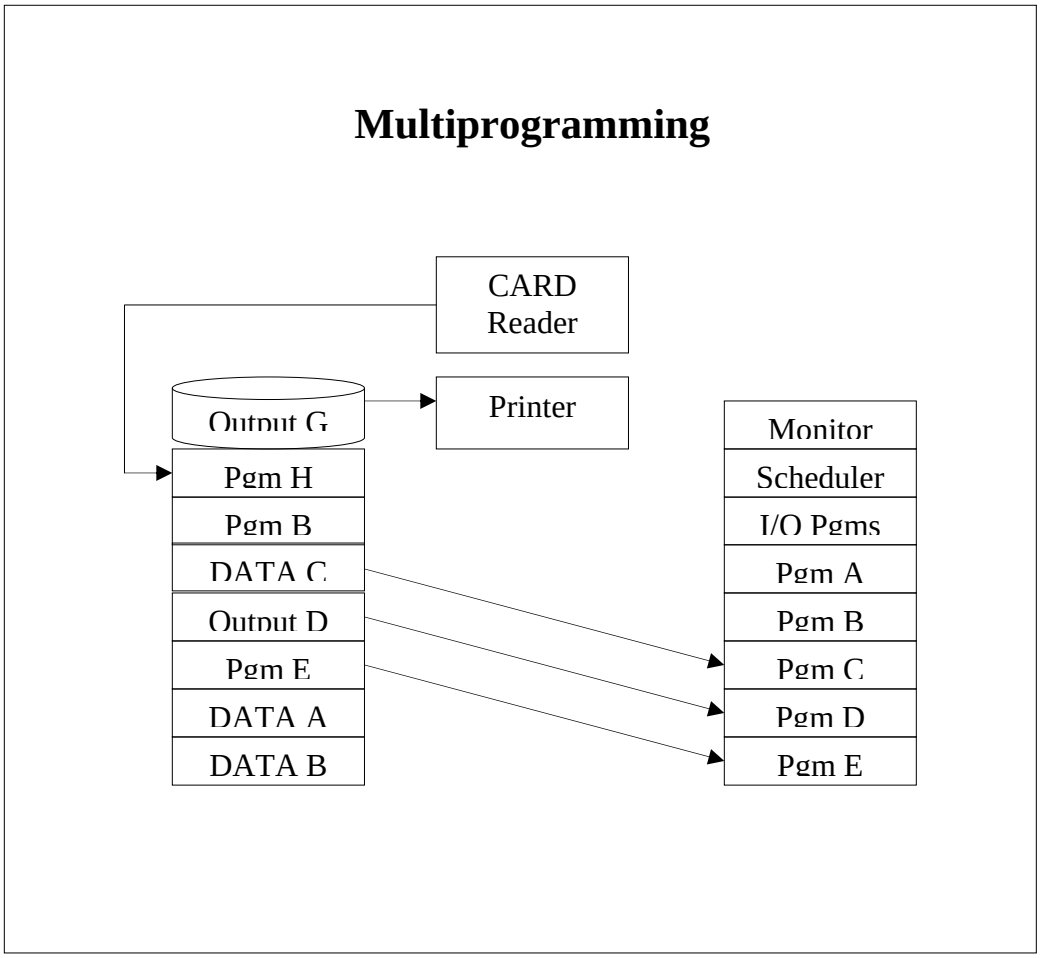



SYSTEM/370
ES/9000
MAINFRAMES
3174
COMM.
Contr. Unit
Modem
3745
Tape
Controller
Modem
Modem
3490
3490
3490
3490
Mode
TC
9343
3390
3270

Large Computer System




OPERATING SYSTEM
O/S
Open a File
Edit
Store
Compile
Execute
M/C Language
H/W
Compiler
Logout
Login


Multiprogramming
CARD Reader
Printer
Output G
Pgm H
Pgm B
DATA C
Output D
Pgm E
DATA A
DATA B
Monitor
Scheduler
I/O Pgms
Pgm A
Pgm B
Pgm C
Pgm D
Pgm E