

# Operaciones de Entrada/Salida

## Plan de Becarios de Supercómputo y Centro de Datos

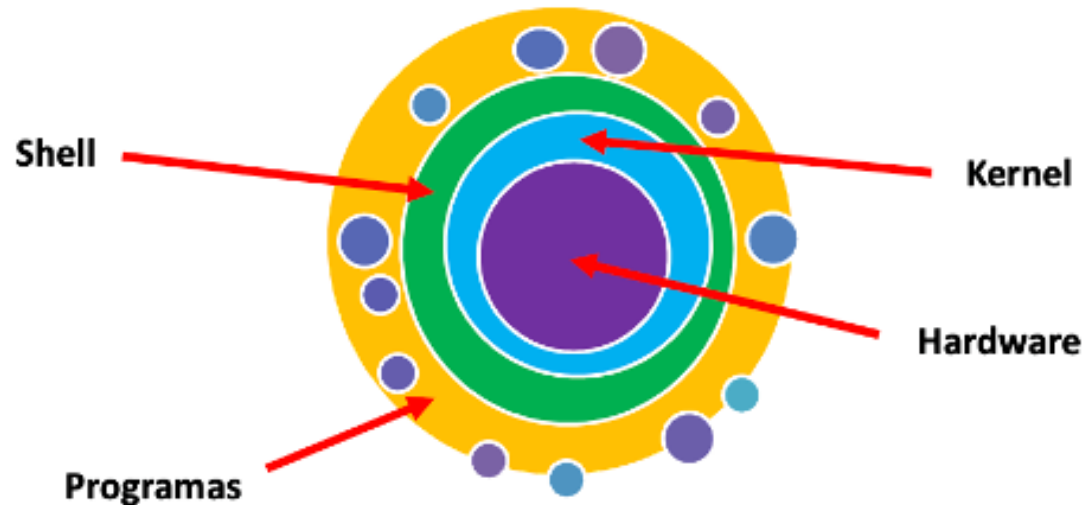
M. en I. Yolanda Flores  
[yoli@unam.mx](mailto:yoli@unam.mx)

Departamento de Supercómputo,  
DGTIC, UNAM.



# Estructura del sistema operativo UNIX

- Para entender cómo funciona UNIX, se debe entender su estructura.



## Relaciones:

- Usuario: Solicita ejecución de los programas a través de la línea de comandos.
- Shell: Interpreta la línea de comandos y solicita la ejecución.
- Llamadas al sistema: Interfaz entre un proceso y el sistema operativo.
- Sistema de archivos:
  - Entrada/Salida
  - Control de procesos

# Filosofía de Unix

This is the Unix philosophy: Write programs that do one thing and do it well. Write programs to work together. Write programs to handle text streams, because that is a universal interface.

Douglas McIlroy

# Operaciones de Entrada/Salida

- **Archivos**

¡todo es un archivo!

## Byte Stream

- Un archivo se presenta como una secuencia no estructurada de bytes.
- Al sistema operativo no le interesa el contenido (lo único que ve son bytes).

- Un proceso aceptará entrada de cualquier fuente (archivo) y enviará la salida hacia cualquier destino (archivo).





# Tabla de Archivos del Kernel

- Para cada proceso el kernel mantiene una tabla de archivos abiertos (donde el archivo puede ser cualquier dispositivo de entrada/salida).
- Esa tabla tiene entradas que corresponden a los descriptores de archivo (fd), los cuales son números enteros obtenidos por medio de la llamada al sistema open().
- Cada proceso en UNIX tiene asociados tres archivos abiertos por defecto:
  - STDIN entrada estándar, fd = 0
  - STDOUT salida estándar, fd = 1
  - STDERR salida de error estándar, fd = 2

```
#include <stdio.h>
#include <stdlib.h>
int main(){
    FILE *fd;
    fd = fopen ("archivo", "r" );

    if (fd==NULL) {
        fputs ("Error", stderr);
        exit (1);
    }
    printf("FD del archivo es
    %d\n", fileno(fd));
    fclose ( fd );
    return 0;
}
```

Significado	Forma	Sintaxis Bourne shell	Sintaxis C shell
stdin de archivo	ND	prog < archivo	
stdout a archivo	D	prog > archivo	
stderr a archivo	D	prog 2> archivo	
stdout y stderr a archivo	D	prog > archivo 2>&1	prog >& archivo
stdin de fuente actual hasta delimitador.	ND	prog << delimitador	
Añade stdout a archivo	ND	prog >> archivo	
Añade stderr a archivo	ND	prog 2>> archivo	
Añade stdout y stderr a archivo	ND	prog >> archivo 2>&1	prog >>& archivo
stdout de prog1 a stdin de prog2	-	prog1   prog2	
stdout y stderr de prog1 a stdin de prog2	-	prog1 2>&1  prog2	prog1  & archivo
Bifurcación de stdout.	D	prog   tee archivo	
Bifurcación de stdout.	ND	prog   tee -a archivo	