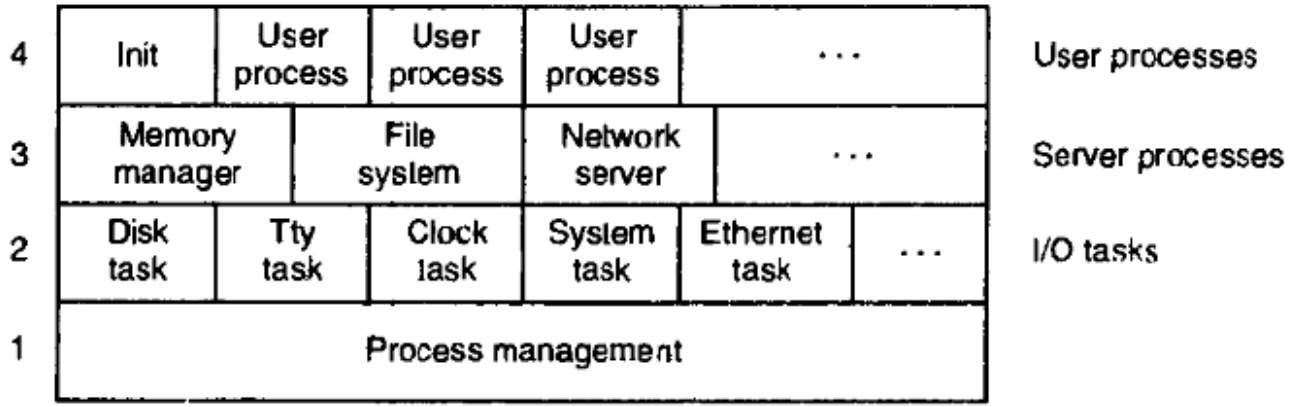
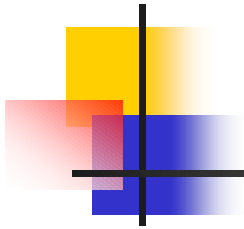
Componentes de un S.O.

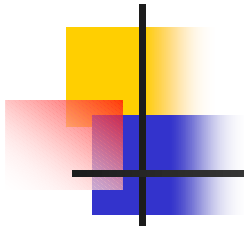
- En Particular Minix está dividido en módulos
  - 1) MicroKernel
  - 2) Tareas E/S
  - 3) Procesos Servidores
  - 4) Procesos de Usuarios
- Son módulos independientes
- Para arquitectura FLYNN - SISD





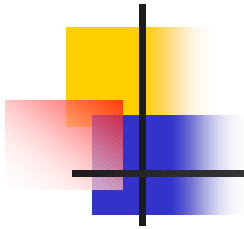
Funciones de un S.O.

- Administración de Procesador
  - Cambio de Estados para procesos
  - Politica de Asignación de procesador
  - Política de Ordenación de Colas
- Administración de Memoria
  - Simple contigua
  - Paginada / Segmentada
- Administración de la Información
  - Layout en disco
  - Funciones de acceso al sistema de archivos
    - Transparencia
- Administración de Reloj del sistema
  - Fecha, Hora



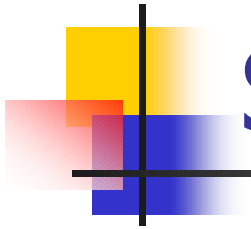
Funciones de un S.O.

- Administración de Recursos
  - Dedicados
    - Manejo de Deadlocks
  - Compartidos
- Comunicación entre procesos
  - Send/receive
    - Bloqueantes / No
    - Con Buffer / Sin
  - Share memory
  - Pipes
  - Sockets
- Sincronización entre procesos
  - Semáforos, Regiones Críticas, Monitores



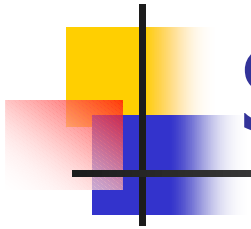
System Calls - Servicios

- [PROC] Administración de Procesos
  - fork, waitpid, wait, exit, execve, ...
- [IPC] Señales
  - kill, pause, alarm, sigaction, ...
- [FS] Administración de Archivos
  - creat, mknod, open, close, read, write, dup, pipe, ...
- [FS] Administración del Sistema de Archivos
  - mkdir, rmdir, link, mount, umount, chdir, chroot, ...
- [FS] Protección
  - chmod, getuid, setuid, getgid, setgid, chown, ...
- [TIME] Administración de Tiempo
  - time, stime, utime, times, ...



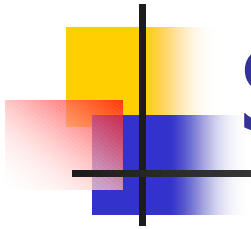
Sistema Operativo en MIMD

- Existen dos tipos de arquitecturas MIMD
  - Fuertemente acopladas
    - Multiprocesadores
  - Debilmente acopladas
    - Multicomputadores
- Clementina II - SGI (MIMD – FA)
  - Arquitectura
    - 40 procesadores
    - Inteconectados por Cray-links (Hipercubo grado 3)
    - Memoria distribuida (NUMA)
    - Función de Ruteo
  - Sistema Operativo IRIX
    - Tiene share-memory y permite el uso de Threads



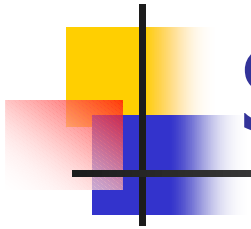
Sistema Operativo en MIMD

- Fenix – SUN Enterprise (MIMD – FA)
  - Arquitectura
    - 16 procesadores
    - Interconectados por Bus
    - Memoria distribuida (UMA)
  - Sistema Operativo SOLARIS
    - Permite el uso de Threads
- Sheldon – Cluster Intel Xeon (MIMD – DA)
  - Arquitectura
    - 40 nodos - dual procesador
    - Interconectados por Red Ethernet 1 Gbit
    - Memoria distribuida (NORMA)



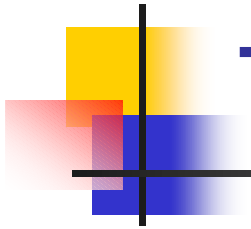
Sistema Operativo en MIMD

- Sheldon – Cluster Intel Xeon (MIMD – DA)
  - Sistema Operativo Linux (Varios)
    - Permite uso de Threads dentro de un nodo.
    - Entre nodo por pasaje de mensaje
    - No hay unica visión de sistema operativo
    - Necesidad de JOB SCHEDULER para asignacion de recursos
    - Necesidad de un FS para todos los nodos.
      - File System de Red
    - Autenticación entre los distintos S.O.
    - No existe Share Memory entre nodos.



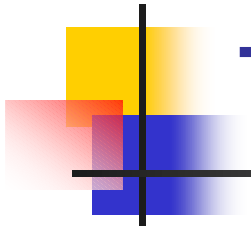
Sistema Operativo en MIMD

- IDEAL – Cluster (MIMD – DA)
  - Sistema Operativo Distribuido
    - Visión única de cola de Procesos
    - Visión única de File System
    - Visión única de Memoria
    - Transparencia en la ubicación de Recursos.
    - Transparencia en la ejecución de Procesos.
    - Migración de Procesos entre los nodos
    - Permite uso de Threads.
  - Módulos cooperativos para brindar servicio
  - Coordinación de módulos
    - Distribuido / Centralizado
  - Coherencia



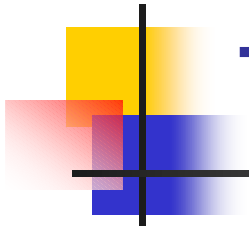
Temas de Implementación

- Comunicación entre nodos
  - Primitivas Send/Receive
    - Conexión y Confiabilidad
    - Niveles de conectividad y confiabilidad (ACKs)
  - Función de Ruteo
  - Tipo de Medio de Transmisión
- Identificación de nodos
  - Estaticos / Con cambios
- Identificación de servicios
  - Estaticos / Con cambios
- Stacks ISO / TCP-IP
- Modelo Cliente/Servidor
  - Send / Receive / Accept



Temas de Implementación

- Uso de procesadores remotos
  - Ejecución Asincrónica
    - Cliente/Servidor
  - Ejecución Sincrónica
    - Remote Procedure Call
- Remote Procedure Call
  - Simula la llamada a un procedimiento remoto como si fuera local.
  - Exiten herramientas que generan el código fuente
    - RPCGEN (XDR, SRC) → (SRC)
  - Se generan los stubs cliente y servidor.
  - Existe binding dinámico y registración del servidor.

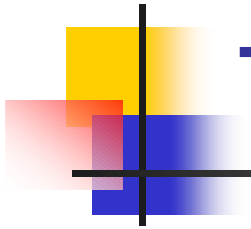


Temas de implementación

- Remote Procedure Call (Camino Critico)
  - El procedimiento cliente llama al stub cliente de manera transparente. Usando Stack.
  - El stub cliente arma el mensaje y se lo envía al kernel.
  - El kernel realiza el send del mensaje al kernel de la máquina remota.
  - El kernel remoto le da el mensaje al stub del server
  - El stub del server desempaqueta los parámetros y se los pasa al server. Usan Stack.
  - El server propiamente dicho realiza su trabajo y retorna un resultado al stub.
  - El stub del server empaqueta el valor retornado y se lo manda al kernel.
  - El kernel remoto envía el mensaje al kernel del cliente.
  - El kernel del cliente sube el mensaje al stub del cliente.
  - El stub cliente desempaqueta el resultado y se lo pasa al cliente.

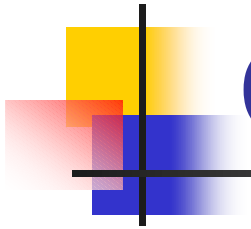
Temas de implementación

- Remote Procedure Call (Semántica de Fallas)
  - El Cliente no puede ubicar al servidor
    - EXCEPCIÓN
  - Se pierde el msg de requerimiento del cliente al servidor
    - Retransmisión al no recibir ACK usando TIMER
  - El msg de respuesta del servidor se pierde
    - Diferenciar esta falla con la anterior. (nro de secuencia)
  - El servidor se cae luego de recibir el requerimiento
    - A) Recibe y procesa, se cae antes de enviar la respuesta
    - B) Se cae antes de procesar el pedido
      - Semántica: At Lest Once – At most Once – Exactly Once
  - El Cliente se cae
    - Huerfanos :
      - Reencarnación – Reencarnación Suave – Expiración Exterminación.



Temas de Diseño de S.O.

- Transparencia
  - De Locación / De Migración / De Réplica
  - De Concurrencia / De Paralelismo
- Fexibilidad
  - Monolitico / Microkernel
- Confiabilidad
- Performance
  - Metricas
    - Tiempo de Respuesta / Rendimiento
    - Uso del Sistema / Capacidad consumida de Red
- Escalabilidad
  - NFS no es escalable



Consultas ?

- Arquitecturas MIMD
  - Tipos / Performance
- Sistemas Operativos Distribuidos
  - Modulos
- Sistemas Distribuidos
  - Servicios
- Modelo Cliente-Servidor
  - RPC
- Varias.....