

INTRODUCCION A LOS SISTEMAS DISTRIBUIDOS

- INTRODUCCION A LOS SISTEMAS DISTRIBUIDOS
- VENTAJAS DE LOS SISTEMAS DISTRIBUIDOS CON RESPECTO A LOS CENTRALIZADOS
- VENTAJAS DE LOS SISTEMAS DISTRIBUIDOS CON RESPECTO A LAS PC INDEPENDIENTES
- DESVENTAJAS DE LOS SISTEMAS DISTRIBUIDOS
- CONCEPTOS DE HARDWARE
- MULTIPROCESADORES CON BASE EN BUSES
- MULTIPROCESADORES CON CONMUTADOR
- MULTICOMPUTADORAS CON BASE EN BUSES
- MULTICOMPUTADORAS CON CONMUTADOR
- CONCEPTOS DE SOFTWARE

INTRODUCCION A LOS SISTEMAS DISTRIBUIDOS

1

INTRODUCCION A LOS SISTEMAS DISTRIBUIDOS

- DESDE EL INICIO DE LA ERA DE LA COMPUTADORA MODERNA (1945), HASTA CERCA DE 1985, SOLO SE CONOCIA LA COMPUTACION CENTRALIZADA.
- A PARTIR DE LA MITAD DE LA DECADA DE LOS OCHENTAS APARECEN DOS AVANCES TECNOLOGICOS FUNDAMENTALES:
 - ◆ DESARROLLO DE MICROPROCESADORES PODEROSOS Y ECONOMICOS:
 - ARQUITECTURAS DE 8, 16, 32 Y 64 BITS.
 - ◆ DESARROLLO DE REDES DE AREA LOCAL (LAN) DE ALTA VELOCIDAD:
 - POSIBILIDAD DE CONECTAR CIENTOS DE MAQUINAS A VELOCIDADES DE TRANSFERENCIA DE MILLONES DE BITS POR SEGUNDO (Mb/SEG).

INTRODUCCION A LOS SISTEMAS DISTRIBUIDOS

4

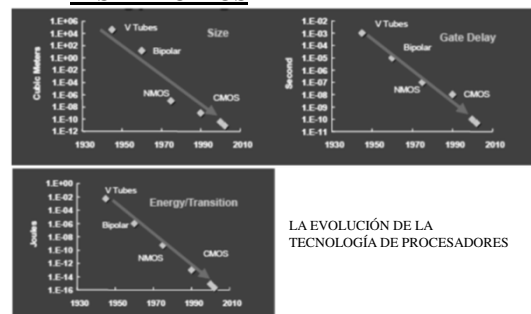
INTRODUCCION A LOS SISTEMAS DISTRIBUIDOS

- SISTEMAS OPERATIVOS DE REDES
- SISTEMAS REALMENTE DISTRIBUIDOS
- SISTEMAS DE MULTIPROCESADOR CON TIEMPO COMPARTIDO
- ASPECTOS DEL DISEÑO
- TRANSPARENCIA
- FLEXIBILIDAD
- CONFIABILIDAD
- DESEMPEÑO
- ESCALABILIDAD

INTRODUCCION A LOS SISTEMAS DISTRIBUIDOS

2

INTRODUCCION A LOS SISTEMAS DISTRIBUIDOS



INTRODUCCION A LOS SISTEMAS DISTRIBUIDOS

5

INTRODUCCION A LOS SISTEMAS DISTRIBUIDOS

INTRODUCCION A LOS SISTEMAS DISTRIBUIDOS

3

INTRODUCCION A LOS SISTEMAS DISTRIBUIDOS



INTRODUCCION A LOS SISTEMAS DISTRIBUIDOS

6

INTRODUCCION A LOS SISTEMAS DISTRIBUIDOS

- APARECEN LOS **SISTEMAS DISTRIBUIDOS**, EN CONTRASTE CON LOS **SISTEMAS CENTRALIZADOS**.
- LOS **SISTEMAS DISTRIBUIDOS** NECESITAN UN **SOFTWARE DISTINTO** AL DE LOS **SISTEMAS CENTRALIZADOS**.
- LOS S. O. PARA **SISTEMAS DISTRIBUIDOS** HAN TENIDO **IMPORTANTES DESARROLLOS** PERO TODAVIA EXISTE UN **LARGO CAMINO POR RECORRER**.
- LOS USUARIOS PUEDEN **ACCEDER** A UNA GRAN VARIEDAD DE RECURSOS COMPUTACIONALES:
 - ◆ DE **HARDWARE** Y DE **SOFTWARE**.
 - ◆ **DISTRIBUIDOS** ENTRE UN GRAN N° DE **SISTEMAS COMPUTACIONALES CONECTADOS**.
- UN IMPORTANTE ANTECEDENTE DE LAS REDES DE COMPUTADORAS LO CONSTITUYE **ARPANET**, INICIADA EN 1968 EN LOS EE. UU.

VENTAJAS DE LOS SISTEMAS DISTRIBUIDOS CON RESPECTO A LOS CENTRALIZADOS

- LOS **SISTEMAS DISTRIBUIDOS** GENERALMENTE TIENEN EN **POTENCIA** UNA PROPORCION **PRECIO / DESEMPEÑO** MUCHO MEJOR QUE LA DE UN UNICO SISTEMA **CENTRALIZADO**.
- ALGUNOS AUTORES DISTINGUEN ENTRE:
 - ◆ **SISTEMAS DISTRIBUIDOS**: ESTAN DISEÑADOS PARA QUE MUCHOS USUARIOS TRABAJEN EN FORMA **CONJUNTA**.
 - ◆ **SISTEMAS PARALELOS**: ESTAN DISEÑADOS PARA LOGRAR LA **MAXIMA RAPIDEZ** EN UN UNICO PROBLEMA.
- EN GENERAL SE CONSIDERAN **SISTEMAS DISTRIBUIDOS**, EN SENTIDO AMPLIO, A LOS SISTEMAS EN QUE:
 - ◆ EXISTEN VARIAS CPU CONECTADAS ENTRE SI.
 - ◆ LAS DISTINTAS CPU TRABAJAN DE MANERA **CONJUNTA**.

VENTAJAS DE LOS SISTEMAS DISTRIBUIDOS CON RESPECTO A LOS CENTRALIZADOS

VENTAJAS DE LOS SISTEMAS DISTRIBUIDOS CON RESPECTO A LOS CENTRALIZADOS

- CIERTAS APLICACIONES SON **DISTRIBUIDAS** EN FORMA **INHERENTE**:
 - ◆ EJ.: SISTEMA DE AUTOMATIZACION DE UNA FABRICA:
 - CONTROLA LOS **ROBOTS** Y **MAQUINAS** EN LA LINEA DE MONTAJE.
 - CADA ROBOT O MAQUINA ES CONTROLADO POR SU **PROPIA COMPUTADORA**.
 - LAS DISTINTAS **COMPUTADORAS** ESTAN **INTERCONECTADAS**.

VENTAJAS DE LOS SISTEMAS DISTRIBUIDOS CON RESPECTO A LOS CENTRALIZADOS

- UNA **RAZON** PARA LA **TENDENCIA** HACIA LA **DESCENTRALIZACIONES** LA **ECONOMIA**.
- **HERB GROSCH** FORMULO LA QUE SE LLAMARIA "LEY DE GROSCH":
 - ◆ EL **PODER** DE **COMPUTO** DE UNA **CPU** ES **PROPORCIONAL** AL CUADRADO DE SU **PRECIO**:
 - SI SE PAGA EL DOBLE SE OBTIENE EL CUADRUPLA DEL **DESEMPEÑO**.
 - ◆ FUE **APLICABLE** EN LOS AÑOS SETENTAS Y OCHENTAS A LA **TECNOLOGIA MAINFRAME**.
 - ◆ **NO ES APLICABLE** A LA **TECNOLOGIA** DEL **MICROPROCESADOR**:
 - LA **SOLUCION MAS EFICAZ** EN CUANTO A **COSTO** ES LIMITARSE A UN GRAN N° DE CPU **BARATOS** REUNIDOS EN UN MISMO SISTEMA.

VENTAJAS DE LOS SISTEMAS DISTRIBUIDOS CON RESPECTO A LOS CENTRALIZADOS

- UNA **VENTAJA POTENCIAL** DE UN SISTEMA **DISTRIBUIDO** ES UNA **MAYOR CONFIABILIDAD**:
 - ◆ AL **DISTRIBUIR** LA **CARGA** DE **TRABAJO** EN MUCHAS **MAQUINAS**, LA **FALLA** DE UNA DE **ELLAS** **NO AFECTARA** A LAS **DEMÁS**:
 - LA **CARGA** DE **TRABAJO** PODRIA **DISTRIBUIRSE**.
 - ◆ SI UNA **MAQUINA** SE **DESCOMPONE**:
 - **SOBREVIVE** EL **SISTEMA** COMO UN **TODOS**.

VENTAJAS DE LOS SISTEMAS DISTRIBUIDOS CON RESPECTO A LOS CENTRALIZADOS

- OTRA VENTAJA IMPORTANTE ES LA POSIBILIDAD DEL CRECIMIENTO INCREMENTAL O POR INCREMENTOS:
 - ◆ PODRIAN AÑADIRSE PROCESADORES AL SISTEMA, PERMITIENDO UN DESARROLLO GRADUAL SEGUN LAS NECESIDADES.
 - ◆ NO SON NECESARIOS GRANDES INCREMENTOS DE POTENCIA EN BREVES LAPROS DE TIEMPO.
 - ◆ SE PUEDE AÑADIR PODER DE COMPUTO EN PEQUEÑOS INCREMENTOS.

VENTAJAS DE LOS SISTEMAS DISTRIBUIDOS CON RESPECTO A LAS PC INDEPENDIENTES

- OTRA IMPORTANTE RAZON ES LOGRAR UNA MEJOR COMUNICACION ENTRE LAS PERSONAS:
 - ◆ EJ.: CORREO ELECTRONICO:
 - POSEE IMPORTANTES VENTAJAS SOBRE EL CORREO POR CARTAS, EL TELEFONO Y EL FAX:
 - VELOCIDAD, DISPONIBILIDAD, GENERACION DE DOCUMENTOS EDITABLES POR PROCESADORES DE TEXTO, ETC.

VENTAJAS DE LOS SISTEMAS DISTRIBUIDOS CON RESPECTO A LAS PC INDEPENDIENTES

VENTAJAS DE LOS SISTEMAS DISTRIBUIDOS CON RESPECTO A LAS PC INDEPENDIENTES

- LA MAYOR FLEXIBILIDAD ES TAMBIEN IMPORTANTE:
 - ◆ LA CARGA DE TRABAJO SE PUEDE DIFUNDIR (DISTRIBUIR) ENTRE LAS MAQUINAS DISPONIBLES EN LA FORMA MAS EFICAZ SEGUN EL CRITERIO ADOPTADO (POR EJ. COSTOS).
 - ◆ LOS EQUIPOS DISTRIBUIDOS PUEDEN NO SER SIEMPRE PC:
 - SE PUEDEN ESTRUCTURAR SISTEMAS CON GRUPOS DE PC Y DE COMPUTADORAS COMPARTIDAS, DE DISTINTA CAPACIDAD.

VENTAJAS DE LOS SISTEMAS DISTRIBUIDOS CON RESPECTO A LAS PC INDEPENDIENTES

- SATISFACEN LA NECESIDAD DE MUCHOS USUARIOS DE COMPARTIR CIERTOS DATOS:
 - ◆ EJ.: SISTEMA DE RESERVAS DE LINEAS AEREAS.
- TAMBIEN CON LOS SISTEMAS DISTRIBUIDOS SE PUEDEN COMPARTIR OTROS RECURSOS COMO PROGRAMAS Y PERIFERICOS COSTOSOS:
 - ◆ EJ.: IMPRESORAS LASER COLOR, EQUIPOS DE FOTOCOMPOSICION, DISPOSITIVOS DE ALMACENAMIENTO MASIVO (EJ.: CAJAS OPTICAS), ETC.

DESVENTAJAS DE LOS SISTEMAS DISTRIBUIDOS

DESVENTAJAS DE LOS SISTEMAS DISTRIBUIDOS

- EL PRINCIPAL PROBLEMA ES EL **SOFTWARE**:
 - ◆ EL DISEÑO, IMPLANTACION Y USO DEL SOFTWARE DISTRIBUIDO PRESENTA **NUMEROSOS INCONVENIENTES**.
- LOS **PRINCIPALES INTERROGANTES** SON LOS SIGUIENTES:
 - ◆ ¿**QUE TIPO** DE S. O., LENGUAJE DE PROGRAMACION Y APLICACIONES SON ADECUADOS PARA ESTOS SISTEMAS?
 - ◆ ¿**CUANTO DEBEN SABER** LOS USUARIOS DE LA DISTRIBUCION?
 - ◆ ¿**QUE TANTO DEBE HACER** EL SISTEMA Y QUE TANTO DEBEN HACER LOS USUARIOS?
- LA **RESPUESTA A ESTOS INTERROGANTES NO ES UNIFORME** ENTRE LOS ESPECIALISTAS:
 - ◆ EXISTE UNA GRAN **DIVERSIDAD DE CRITERIOS** Y DE **INTERPRETACIONES** AL RESPECTO.

CONCEPTOS DE HARDWARE

- **TODOS LOS SISTEMAS DISTRIBUIDOS CONSTAN DE VARIAS CPU, ORGANIZADAS** DE DIVERSAS FORMAS, ESPECIALMENTE RESPECTO DE:
 - ◆ LA **FORMA DE INTERCONECTARLAS** ENTRE SI.
 - ◆ LOS **ESQUEMAS DE COMUNICACION** UTILIZADOS.

DESVENTAJAS DE LOS SISTEMAS DISTRIBUIDOS

- OTRO **PROBLEMA POTENCIAL** TIENE QUE VER CON LAS **REDES DE COMUNICACIONES**:
 - ◆ SE DEBEN CONSIDERAR PROBLEMAS DEBIDOS A **PERDIDAS** DE MENSAJES, **SATURACION** EN EL TRAFICO, **EXPANSION**, ETC.
- EL HECHO DE QUE SEA **FACIL COMPARTIR LOS DATOS** ES UNA **VENTAJA** PERO SE PUEDE CONVERTIR EN UN **GRAN PROBLEMA**:
 - ◆ LA **SEGURIDAD** DEBE ORGANIZARSE ADECUADAMENTE.
- EN GENERAL SE CONSIDERA QUE LAS **VENTAJAS SUPERAN A LAS DESVENTAJAS**, SI ESTAS ULTIMAS SE ADMINISTRAN SERIAMENTE.

CONCEPTOS DE HARDWARE

- EXISTEN **DIVERSOS ESQUEMAS DE CLASIFICACION** PARA LOS SISTEMAS DE COMPUTOS CON VARIAS CPU:
 - ◆ UNO DE LOS MAS CONOCIDOS ES LA **TAXONOMIA DE FLYNN**:
 - CONSIDERA COMO **CARACTERISTICAS ESENCIALES** EL N° DE **FLUJO DE INSTRUCCIONES** Y EL N° DE **FLUJOS DE DATOS**.
 - LA CLASIFICACION INCLUYE EQUIPOS **SISD, SIMD, MISD Y MIMD**.
- **SISD** (SINGLE INSTRUCTION SINGLE DATA: **UN FLUJO DE INSTRUCCIONES Y UN FLUJO DE DATOS**):
 - ◆ POSEEN UN **UNICO PROCESADOR**.

CONCEPTOS DE HARDWARE

CONCEPTOS DE HARDWARE

- **SIMD** (SINGLE INSTRUCTION MULTIPLE DATA: **UN FLUJO DE INSTRUCCIONES Y VARIOS FLUJOS DE DATOS**):
 - ◆ SE REFIERE A **ORDENAR PROCESADORES** CON UNA **UNIDAD DE INSTRUCCION** QUE:
 - **BUSCA** UNA INSTRUCCION.
 - **INSTRUYE A VARIAS UNIDADES DE DATOS** PARA QUE LA LLEVEN A CABO EN **PARALELO**, C / U CON SUS PROPIOS DATOS.
 - ◆ SON **UTILES** PARA LOS COMPUTOS QUE REPITEN LOS **MISMOS CALCULOS** EN **VARIOS CONJUNTOS DE DATOS**.

CONCEPTOS DE HARDWARE

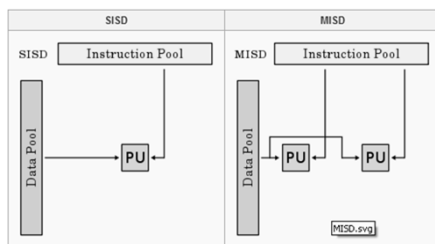
- MISD (MULTIPLE INSTRUCTION SINGLE DATA: UN FLUJO DE VARIAS INSTRUCCIONES Y UN SOLO FLUJO DE DATOS):
 - ◆ NO SE PRESENTA EN LA PRACTICA.
- MIMD (MULTIPLE INSTRUCTION MULTIPLE DATA: UN GRUPO DE COMPUTADORAS INDEPENDIENTES, C / U CON SU PROPIO CONTADOR DEL PROGRAMA, PROGRAMA Y DATOS):
 - ◆ *TODOS LOS SISTEMAS DISTRIBUIDOS SON DE ESTE TIPO.*

CONCEPTOS DE HARDWARE

- UN AVANCE SOBRE LA CLASIFICACION DE FLYNN INCLUYE LA DIVISION DE LAS COMPUTADORAS MIMD EN DOS GRUPOS:
 - ◆ MULTIPROCESADORES: POSEEN MEMORIA COMPARTIDA:
 - LOS DISTINTOS PROCESADORES COMPARTEN EL MISMO ESPACIO DE DIRECCIONES VIRTUALES.
 - ◆ MULTICOMPUTADORAS: NO POSEEN MEMORIA COMPARTIDA:
 - EJ.: GRUPO DE PC CONECTADAS MEDIANTE UNA RED.

CONCEPTOS DE HARDWARE

TAXONOMIA DE FLYNN

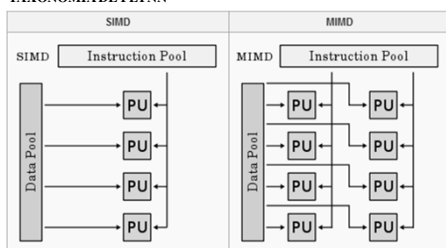


CONCEPTOS DE HARDWARE

- C / U DE LAS CATEGORIAS INDICADAS SE PUEDE CLASIFICAR SEGUN LA ARQUITECTURA DE LA RED DE INTERCONEXIONEN:
 - ◆ ESQUEMA DE BUS:
 - EXISTE UNA SOLA RED, BUS, CABLE U OTRO MEDIO QUE CONECTA TODAS LAS MAQUINAS:
 - EJ.: LA TELEVISION POR CABLE.
 - ◆ ESQUEMA CON CONMUTADOR:
 - NO EXISTE UNA SOLA COLUMNA VERTEBRAL DE CONEXION:
 - HAY MULTIPLES CONEXIONES Y VARIOS PATRONES DE CONEXIONADO.
 - LOS MENSAJES DE MUEVEN A TRAVES DE LOS MEDIOS DE CONEXION.
 - SE DECIDE EXPLICITAMENTE LA CONMUTACION EN CADA ETAPA PARA DIRIGIR EL MENSAJE A LO LARGO DE UNO DE LOS CABLES DE SALIDA.
 - EJ.: EL SISTEMA MUNDIAL TELEFONICO PUBLICO.

CONCEPTOS DE HARDWARE

TAXONOMIA DE FLYNN

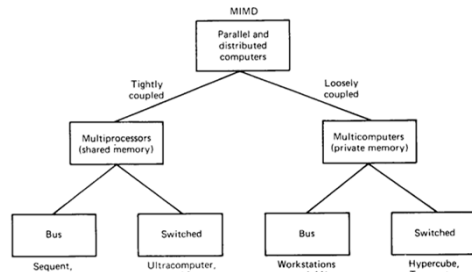


CONCEPTOS DE HARDWARE

- OTRO ASPECTO DE LA CLASIFICACION CONSIDERA EL ACOPLAMIENTO ENTRE LOS EQUIPOS:
 - ◆ SISTEMAS FUERTEMENTE ACOPLADOS:
 - EL RETRASO AL ENVIAR UN MENSAJE DE UNA COMPUTADORA A OTRA ES CORTO Y LA TASA DE TRANSMISION ES ALTA.
 - GENERALMENTE SE LOS UTILIZA COMO SISTEMAS PARALELOS.
 - ◆ SISTEMAS DEBILMENTE ACOPLADOS:
 - EL RETRASO DE LOS MENSAJES ENTRE LAS MAQUINAS ES GRANDE Y LA TASA DE TRANSMISION ES BAJA.
 - GENERALMENTE SE LOS UTILIZA COMO SISTEMAS DISTRIBUIDOS.
- GENERALMENTE LOS MULTIPROCESADORES ESTAN MAS FUERTEMENTE ACOPLADOS QUE LAS MULTICOMPUTADORAS.

CONCEPTOS DE HARDWARE

UNA TAXONOMIA DIFERENTE



INTRODUCCION A LOS SISTEMAS DISTRIBUIDOS

31

MULTIPROCESADORES CON BASE EN BUSES

- PARA GRABAR EL PROCEDIMIENTO ES SIMILAR.
- SOLO EXISTE UNA MEMORIA, LA CUAL PRESENTA LA PROPIEDAD DE LA COHERENCIA:
 - ◆ LAS MODIFICACIONES HECHAS POR UNA CPU SE REFLEJAN DE INMEDIATO EN LAS SUBSIGUIENTES LECTURAS DE LA MISMA O DE OTRA CPU.

INTRODUCCION A LOS SISTEMAS DISTRIBUIDOS

34

MULTIPROCESADORES CON BASE EN BUSES

INTRODUCCION A LOS SISTEMAS DISTRIBUIDOS

32

MULTIPROCESADORES CON BASE EN BUSES

- EL PROBLEMA DE ESTE ESQUEMA ES QUE EL BUS TIENDE A SOBRECARGARSE Y EL RENDIMIENTO A DISMINUIR DRASTICAMENTE:
 - ◆ LA SOLUCION ES AÑADIR UNA MEMORIA CACHE DE ALTA VELOCIDAD ENTRE LA CPU Y EL BUS:
 - EL CACHE GUARDA LAS PALABRAS DE ACCESO RECIENTE.
 - TODAS LAS SOLICITUDES DE LA MEMORIA PASAN A TRAVES DEL CACHE.
 - SI LA PALABRA SOLICITADA SE ENCUENTRA EN EL CACHE:
 - EL CACHE RESPONDE A LA CPU.
 - NO SE HACE SOLICITUD ALGUNA AL BUS.
 - ◆ SI EL CACHE ES LO BASTANTE GRANDE:
 - LA TASA DE ENCUENTROS SERA ALTA Y LA CANTIDAD DE TRAFICO EN EL BUS POR CADA CPU DISMINUIRA DRASTICAMENTE:
 - PERMITE INCREMENTAR EL N° DE CPU.

INTRODUCCION A LOS SISTEMAS DISTRIBUIDOS

35

MULTIPROCESADORES CON BASE EN BUSES

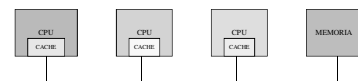
- CONSTAN DE CIERTO N° DE CPU CONECTADAS A UN BUS COMUN, JUNTO CON UN MODULO DE MEMORIA.
- UN BUS TÍPICO POSEE AL MENOS:
 - ◆ 32 LINEAS DE DIRECCIONES.
 - ◆ 32 LINEAS DE DATOS.
 - ◆ 30 LINEAS DE CONTROL.
- TODOS LOS ELEMENTOS PRECEDENTES OPERAN EN PARALELO.
- PARA LEER UNA PALABRA DE MEMORIA, UNA CPU:
 - ◆ COLOCA LA DIRECCION DE LA PALABRA DESEADA EN LAS LINEAS DE DIRECCIONES DEL BUS.
 - ◆ COLOCA UNA SEÑAL EN LAS LINEAS DE CONTROL ADECUADAS PARA INDICAR QUE DESEA LEER.
 - ◆ LA MEMORIA RESPONDE Y COLOCA EL VALOR DE LA PALABRA EN LAS LINEAS DE DATOS PARA PERMITIR LA LECTURA DE ESTA POR PARTE DE LA CPU SOLICITANTE.

INTRODUCCION A LOS SISTEMAS DISTRIBUIDOS

33

MULTIPROCESADORES CON BASE EN BUSES

MULTIPROCESADORES CON BASE EN UN BUS



INTRODUCCION A LOS SISTEMAS DISTRIBUIDOS

36

MULTIPROCESADORES CON BASE EN BUSES

- UN IMPORTANTE **PROBLEMA** DEBIDO AL USO DE CACHES ES EL DE LA **INCOHERENCIA DE LA MEMORIA**:
 - ◆ SUPONGAMOS QUE LAS CPU **A Y B** LEEN LA MISMA PALABRA DE MEMORIA EN SUS RESPECTIVOS **CACHES**.
 - ◆ **A ESCRIBE** SOBRE LA PALABRA.
 - ◆ CUANDO **B** LEE ESA PALABRA, OBTIENE UN **VALOR** ANTERIOR Y NO EL VALOR RECIENTE ACTUALIZADO POR A.

MULTIPROCESADORES CON BASE EN BUSES

- [VER ANIMACIÓN DEMOSTRATIVA](#).

MULTIPROCESADORES CON BASE EN BUSES

- UNA **SOLUCION** CONSISTE EN LO SIGUIENTE:
 - ◆ **DISEÑAR** LAS CACHE DE TAL FORMA QUE CUANDO UNA PALABRA SEA ESCRITA AL CACHE, TAMBIEN SEA ESCRITA A LA MEMORIA.
 - ◆ A ESTO SE DENOMINA **CACHE DE ESCRITURA**.
 - ◆ **NO CAUSA TRAFICO** EN EL BUS EL USO DE CACHE PARA LA LECTURA.
 - ◆ **SI CAUSA TRAFICO** EN EL BUS:
 - EL NO USO DE CACHE PARA LA LECTURA.
 - **TODA LA ESCRITURA**.

MULTIPROCESADORES CON CONMUTADOR

MULTIPROCESADORES CON BASE EN BUSES

- SI TODOS LOS CACHES REALIZAN UN **MONITOREO CONSTANTE DEL BUS**:
 - ◆ CADA VEZ QUE UN CACHE **OBSERVA** UNA ESCRITURA A UNA DIRECCION DE MEMORIA PRESENTE EN EL:
 - PUEDE **ELIMINAR ESE DATO** O **ACTUALIZARLO** EN EL CACHE CON EL NUEVO VALOR.
 - ◆ ESTOS CACHES SE DENOMINAN **CACHES MONITORES**.
- UN **DISEÑO CON CACHES MONITORES Y DE ESCRITURA ES COHERENTE E INVISIBLE PARA EL PROGRAMADOR**:
 - ◆ ES MUY UTILIZADO EN MULTIPROCESADORES BASADOS EN BUSES.

MULTIPROCESADORES CON CONMUTADOR

- EL ESQUEMA DE MULTIPROCESADORES CON BASE EN BUSES RESULTA **APROPIADO** PARA HASTA APROXIMADAMENTE **64 PROCESADORES**.
- **PARA SUPERAR** ESTA CIFRA ES NECESARIO UN **METODO DISTINTO DE CONEXION** ENTRE PROCESADORES (CPU) Y MEMORIA.

MULTIPROCESADORES CON CONMUTADOR

- UNA POSIBILIDAD ES DIVIDIR LA MEMORIA EN MODULOS Y CONECTARLOS A LAS CPU CON UN CONMUTADOR DE CRUCETA (CROSS-BAR SWITCH):
 - CADA CPU Y CADA MEMORIA TIENE UNA CONEXION QUE SALE DE EL.
 - EN CADA INTERSECCION ESTA UN CONMUTADOR DEL PUNTO DE CRUCE (CROSSPOINT SWITCH) ELECTRONICO QUE EL HARDWARE PUEDE ABRIR Y CERRAR:
 - CUANDO UNA CPU DESEA TENER ACCESO A UNA MEMORIA PARTICULAR, EL CONMUTADOR DEL PUNTO DE CRUCE QUE LOS CONECTA SE CIERRA MOMENTANEAMENTE.

MULTIPROCESADORES CON CONMUTADOR

- EL N° DE CONMUTADORES DEL ESQUEMA ANTERIOR PUEDE RESULTAR PROHIBITIVO:
 - OTROS ESQUEMAS PRECISAN MENOS CONMUTADORES, POR EJ., LA RED OMEGA:
 - POSEE CONMUTADORES 2 X 2:
 - C / U TIENE 2 ENTRADAS Y 2 SALIDAS.
 - CADA CONMUTADOR PUEDE DIRIGIR CUALQUIERA DE LAS ENTRADAS EN CUALQUIERA DE LAS SALIDAS.
 - ELIGIENDO LOS ESTADOS ADECUADOS DE LOS CONMUTADORES:
 - CADA CPU PODRA TENER ACCESO A CADA MEMORIA.

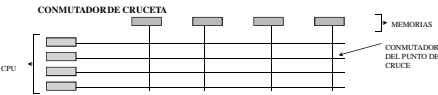
MULTIPROCESADORES CON CONMUTADOR

- LA VIRTUD DEL CONMUTADOR DE CRUCETA ES QUE MUCHAS CPU PUEDEN TENER ACCESO A LA MEMORIA AL MISMO TIEMPO:
 - AUNQUE NO A LA MISMA MEMORIA SIMULTANEAMENTE.
- LO NEGATIVO DE ESTE ESQUEMA ES EL ALTO N° DE CONMUTADORES:
 - PARA n CPU Y n MEMORIAS SE NECESITAN n X n CONMUTADORES.

MULTIPROCESADORES CON CONMUTADOR

- PARA n CPU Y n MEMORIAS SE PRECISAN:
 - n ETAPAS DE CONMUTACION.
 - CADA ETAPA TIENE log₂ n CONMUTADORES PARA UN TOTAL DE n log₂ n CONMUTADORES:
 - ESTE N° ES MENOR QUE n X n DEL ESQUEMA ANTERIOR, PERO SIGUE SIENDO MUY GRANDE PARA n GRANDE.

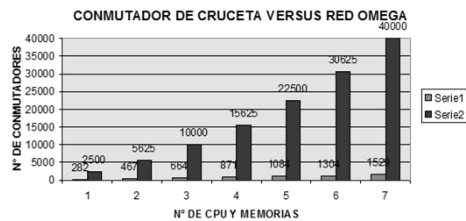
MULTIPROCESADORES CON CONMUTADOR



MULTIPROCESADORES CON CONMUTADOR

n	LOG (n,2)	n * LOG (n,2)	n * n
50	5,64385619	282	2500
75	6,22881869	467	5625
100	6,64385619	664	10000
125	6,96578428	871	15625
150	7,22881869	1084	22500
175	7,45121111	1304	30625
200	7,64385619	1529	40000
1024	10	10240	1048576

MULTIPROCESADORES CON CONMUTADOR



MULTIPROCESADORES CON CONMUTADOR

■ UN PROBLEMA IMPORTANTE EN LA RED OMEGA ES EL RETRASO:

- ◆ EJ.: SI $n = 1024$ EXISTEN SEGUN LA TABLA ANTERIOR:
 - 10 ETAPAS DE CONMUTACION DE LA CPU A LA MEMORIA.
 - 10 ETAPAS PARA QUE LA PALABRA SOLICITADA DE LA MEMORIA REGRESE.
 - SI EL TIEMPO DE EJECUCION DE UNA INSTRUCCION ES DE 20 NSEG.
 - SI UNA SOLICITUD DE LA MEMORIA DEBE RECORRER 20 ETAPAS DE CONMUTACION (10 DE IDA Y 10 DE REGRESO) EN 20 NSEG:
 - EL TIEMPO DE CONMUTACION DEBE SER DE 1 NSEG.
 - EL MULTIPROCESADOR DE 1024 CPU NECESITARA 10240 CONMUTADORES DE 1 NSEG.
 - EL COSTO SERA ALTO.

MULTIPROCESADORES CON CONMUTADOR

ACLARACIONES:

- * SERIE 1: CORRESPONDE A RED OMEGA.
- * SERIE 2: CORRESPONDE A CONMUTADOR DE CRUCETA.
- * LOS VALORES SEÑALADOS EN Nº DE CPU Y MEMORIAS CORRESPONDEN A LA SIGUIENTE EQUIVALENCIA:

1:	50
2:	75
3:	100
4:	125
5:	150
6:	175
7:	200

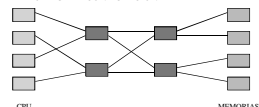
MULTIPROCESADORES CON CONMUTADOR

■ OTRA POSIBLE SOLUCION SON LOS ESQUEMAS SEGUN SISTEMAS JERARQUICOS:

- ◆ CADA CPU TIENE ASOCIADA CIERTA MEMORIA LOCAL.
- ◆ EL ACCESO SERA MUY RAPIDO A LA PROPIA MEMORIA LOCAL Y MAS LENTO A LA MEMORIA DE LAS DEMAS CPU.
- ◆ ESTO SE DENOMINA ESQUEMA O MAQUINA NUMA (ACCESO NO UNIFORME A LA MEMORIA):
 - TIENEN UN MEJOR TIEMPO PROMEDIO DE ACCESO QUE LAS MAQUINAS BASADAS EN REDES OMEGA.
 - LA COLOCACION DE LOS PROGRAMAS Y DATOS EN MEMORIA ES CRITICA PARA LOGRAR QUE LA MAYORIA DE LOS ACCESOS SEAN A LA MEMORIA LOCAL DE CADA CPU.

MULTIPROCESADORES CON CONMUTADOR

RED OMEGA DE CONMUTACION



MULTICOMPUTADORAS CON BASE EN BUSES

MULTICOMPUTADORAS CON BASE EN BUSES

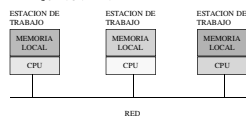
- ES UN ESQUEMA SIN MEMORIA COMPARTIDA.
- CADA CPU TIENE UNA CONEXION DIRECTA CON SU PROPIA MEMORIA LOCAL.
- UN PROBLEMA IMPORTANTE ES LA FORMA EN QUE LAS CPU SE COMUNIQUEN ENTRE SI.
- EL TRAFICO ES SOLO ENTRE UNA CPU Y OTRA:
 - ◆ EL VOLUMEN DE TRAFICO SERA VARIOS ORDENES DE MAGNITUD MENOR QUE SI SE UTILIZARA LA RED DE INTERCONEXION PARA EL TRAFICO CPU - MEMORIA.
- TOPOLOGICAMENTE ES UN ESQUEMA SIMILAR AL DEL MULTIPROCESADOR BASADO EN UN BUS.
- CONSISTE GENERALMENTE EN UNA COLECCION DE ESTACIONES DE TRABAJO EN UNA LAN (RED DE AREA LOCAL).

MULTICOMPUTADORAS CON CONMUTADOR

- CADA CPU TIENE ACCESO DIRECTO Y EXCLUSIVO A SU PROPIA MEMORIA PARTICULAR.
- EXISTEN DIVERSAS TOPOLOGIAS, LAS MAS COMUNES SON LA RETICULA Y EL HIPERCUBO.
- LAS PRINCIPALES CARACTERISTICAS DE LAS RETICULAS SON:
 - ◆ SON FACILES DE COMPRENDER.
 - ◆ SE BASAN EN LAS TARJETAS DE CIRCUITOS IMPRESOS.
 - ◆ SE ADECUAN A PROBLEMAS CON UNA NATURALEZA BIDIMENSIONAL INHERENTE (TEORIA DE GRAFICAS, VISION ARTIFICIAL, ETC.).

MULTICOMPUTADORAS CON BASE EN BUSES

EL DE MULTICOMPUTADORA QUE CONSTA DE ESTACIONES DE TRABAJO EN UNA LAN



MULTICOMPUTADORAS CON CONMUTADOR

- LAS PRINCIPALES CARACTERISTICAS DEL HIPERCUBO SON:
 - ◆ ES UN CUBO n - DIMENSIONAL.
 - ◆ EN UN HIPERCUBO DE DIMENSION 4:
 - SE PUEDE CONSIDERAR COMO DOS CUBOS ORDINARIOS, C / U DE ELLOS CON 8 VERTICES Y 12 ARISTAS.
 - CADA VERTICE ES UN CUBO.
 - CADA ARISTA ES UNA CONEXION ENTRE 2 CPU.
 - SE CONECTAN LOS VERTICES CORRESPONDIENTES DE C / U DE LOS CUBOS.
 - ◆ EN UN HIPERCUBO DE DIMENSION 5:
 - SE DEBERIAN AÑADIR DOS CUBOS CONECTADOS ENTRE SI Y CONECTAR LAS ARISTAS CORRESPONDIENTES EN LAS DOS MITADES, Y ASI SUCESIVAMENTE.

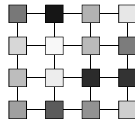
MULTICOMPUTADORAS CON CONMUTADOR

MULTICOMPUTADORAS CON CONMUTADOR

- ◆ EN UN HIPERCUBO DE n DIMENSIONES:
 - CADA CPU TIENE n CONEXIONES CON OTRAS CPU.
 - LA COMPLEJIDAD DEL "CABLEADO" AUMENTA EN PROPORCION LOGARITMICA CON EL TAMAÑO.
 - SOLO SE CONECTAN LOS PROCESADORES VECINOS MAS CERCANOS:
 - MUCHOS MENSAJES DEBEN REALIZAR VARIOS SALTOS ANTES DE LLEGAR A SU DESTINO.
 - LA TRAYECTORIA MAS GRANDE CRECE EN FORMA LOGARITMICA CON EL TAMAÑO:
 - EN LA RETICULA CRECE COMO LA RAIZ CUADRADA DEL N° DE CPU.
 - CON LA TECNOLOGIA ACTUAL YA SE PUEDEN PRODUCIR HIPERCUBOS DE 16384 CPU.

MULTICOMPUTADORAS CON CONMUTADOR

RETICULA

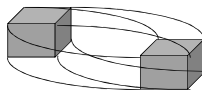


CONCEPTOS DE SOFTWARE

- LA IMPORTANCIA DEL SOFTWARE SUPERA FRECUENTEMENTE A LA DEL HARDWARE.
- LA IMAGEN QUE UN SISTEMA PRESENTA QUEDA DETERMINADA EN GRAN MEDIDA POR EL SOFTWARE DEL S. O. Y NO POR EL HARDWARE.
- LOS S. O. NO SE PUEDEN ENCASILLAR FACILMENTE, COMO EL HARDWARE, PERO SE LOS PUEDE CLASIFICAR EN DOS TIPOS:
 - ◆ DEBILMENTE ACOPLADOS.
 - ◆ FUERTEMENTE ACOPLADOS.
- EL SOFTWARE DEBILMENTE ACOPLADO DE UN SISTEMA DISTRIBUIDO:
 - ◆ PERMITE QUE LAS MAQUINAS Y USUARIOS SEAN INDEPENDIENTES ENTRE SI EN LO FUNDAMENTAL.
 - ◆ FACILITA QUE INTERACTUEN EN CIERTO GRADO CUANDO SEA NECESARIO.
 - ◆ LOS EQUIPOS INDIVIDUALES SE DISTINGUEN FACILMENTE.

MULTICOMPUTADORAS CON CONMUTADOR

HIPERCUBO DE DIMENSION 4



CONCEPTOS DE SOFTWARE

- COMBINANDO LOS DISTINTOS TIPOS DE HARDWARE DISTRIBUIDO CON SOFTWARE DISTRIBUIDO SE LOGRAN DISTINTAS SOLUCIONES:
 - ◆ NO TODAS INTERESAN DESDE EL PUNTO DE VISTA FUNCIONAL DEL USUARIO:
 - ⇒ EJ.: UN MULTIPROCESADOR ES UN MULTIPROCESADOR:
 - NO IMPORTA SI UTILIZA UN BUS CON CACHES MONITORES O UNA RED OMEGA.

CONCEPTOS DE SOFTWARE

SISTEMAS OPERATIVOS DE REDES

SISTEMAS OPERATIVOS DE REDES

- UNA POSIBILIDAD ES EL SOFTWARE DEBILMENTE ACOPLADO EN HARDWARE DEBILMENTE ACOPLADO:
 - ◆ ES UNA SOLUCION MUY UTILIZADA.
 - ◆ EJ.: UNA RED DE ESTACIONES DE TRABAJO CONECTADAS MEDIANTE UNA LAN.

SISTEMAS OPERATIVOS DE REDES

- UNA MEJOR SOLUCION CONSISTE EN UN *SISTEMA DE ARCHIVOS GLOBAL COMPARTIDO*, ACCESIBLE DESDE TODAS LAS ESTACIONES DE TRABAJO:
 - ◆ UNA O VARIAS MAQUINAS SOPORTAN AL SISTEMA DE ARCHIVOS:
 - SON LOS SERVIDORES DE ARCHIVOS.

SISTEMAS OPERATIVOS DE REDES

- CADA USUARIO TIENE UNA ESTACION DE TRABAJO PARA SU USO EXCLUSIVO:
 - ◆ TIENE SU PROPIO S. O.
 - ◆ LA MAYORIA DE LOS REQUERIMIENTOS SE RESUELVEN LOCALMENTE.
 - ◆ ES POSIBLE QUE UN USUARIO SE CONECTE DE MANERA REMOTA CON OTRA ESTACION DE TRABAJO:
 - MEDIANTE UN COMANDO DE "LOGIN REMOTO".
 - SE CONVIERTE LA PROPIA ESTACION DE TRABAJO DEL USUARIO EN UNA **TERMINAL REMOTA** ENLAZADA CON LA MAQUINA REMOTA.
 - LOS COMANDOS SE ENVIAN A LA MAQUINA REMOTA.
 - LA SALIDA DE LA MAQUINA REMOTA SE EXHIBE EN LA PANTALLA LOCAL.

SISTEMAS OPERATIVOS DE REDES

- LOS SERVIDORES DE ARCHIVOS:
 - ◆ **ACEPTAN SOLICITUDES** DE LOS PROGRAMAS DE USUARIOS:
 - LOS PROGRAMAS SE EJECUTAN EN LAS MAQUINAS NO SERVIDORAS, LLAMADAS **CLIENTES**.
 - LAS SOLICITUDES SE EXAMINAN, SE EJECUTAN Y LA RESPUESTA SE ENVIA DE REGRESO.
 - ◆ GENERALMENTE TIENEN UN **SISTEMA JERARQUICO DE ARCHIVOS**.

SISTEMAS OPERATIVOS DE REDES

- ◆ PARA ALTERNAR CON OTRA MAQUINA REMOTA, PRIMERO HAY QUE DESCONECTARSE DE LA PRIMERA:
 - EN CUALQUIER INSTANTE SOLO SE PUEDE UTILIZAR UNA MAQUINA.
- ◆ LAS REDES TAMBIEN DISPONEN DE UN COMANDO DE COPIADO REMOTO DE ARCHIVOS DE UNA MAQUINA A OTRA:
 - REQUIERE QUE EL USUARIO CONOZCA:
 - LA POSICION DE TODOS LOS ARCHIVOS.
 - EL SITIO DONDE SE EJECUTAN TODOS LOS COMANDOS.

SISTEMAS OPERATIVOS DE REDES

- LAS ESTACIONES DE TRABAJO PUEDEN IMPORTAR O MONTAR ESTOS **SISTEMAS DE ARCHIVOS**:
 - ◆ SE INCREMENTAN SUS **SISTEMAS DE ARCHIVOS LOCALES**.
 - ◆ SE PUEDEN MONTAR LOS **SERVIDORES** EN LUGARES DIFERENTES DE SUS RESPECTIVOS **SISTEMAS DE ARCHIVOS**:
 - LAS **RUTAS DE ACCESO** A UN DETERMINADO **ARCHIVO** PUEDEN SER **DIFERENTES** PARA LAS DISTINTAS ESTACIONES.
 - LOS **DISTINTOS CLIENTES** TIENEN UN **PUNTO DE VISTA DISTINTO** DEL SISTEMA DE ARCHIVOS.
 - EL **NOMBRE DE UN ARCHIVO** DEPENDE:
 - DEL LUGAR DESDE EL CUAL SE TIENE ACCESO A EL.
 - DE LA CONFIGURACION DEL SISTEMA DE ARCHIVOS.

SISTEMAS OPERATIVOS DE REDES

- EL S. O. DE ESTE TIPO DE AMBIENTE DEBE:
 - ◆ CONTROLAR LAS ESTACIONES DE TRABAJO EN LO INDIVIDUAL.
 - ◆ CONTROLAR A LOS SERVIDORES DE ARCHIVO.
 - ◆ DEBE ENCARGARSE DE LA COMUNICACION ENTRE LOS SERVIDORES.
- TODAS LAS MAQUINAS PUEDEN EJECUTAR EL MISMO S. O., PERO ESTO NO ES NECESARIO.
- SI LOS CLIENTES Y LOS SERVIDORES EJECUTAN DIVERSOS S. O.:
 - ◆ COMO MINIMO DEBEN COINCIDIR EN EL FORMATO Y SIGNIFICADO DE TODOS LOS MENSAJES QUE PODRIAN INTERCAMBIAR.

SISTEMAS OPERATIVOS DE REDES

- LOS ASPECTOS MAS INTERESANTES SON LOS RELACIONADOS CON:
 - ◆ LA ARQUITECTURA.
 - ◆ EL PROTOCOLO.
 - ◆ LA IMPLANTACION.
- LA ARQUITECTURA DE NFS.
- LA IDEA FUNDAMENTAL ES PERMITIR QUE UNA COLECCION ARBITRARIA DE CLIENTES Y SERVIDORES COMPARTAN UN SISTEMA DE ARCHIVOS COMUN.
- GENERALMENTE TODOS LOS CLIENTES Y SERVIDORES ESTAN EN LA MISMA LAN, PERO ESTO NO ES NECESARIO:
 - ◆ SE PUEDE EJECUTAR NFS EN UNA RED DE AREA AMPLIA.

SISTEMAS OPERATIVOS DE REDES

- ESQUEMAS COMO ESTE SE DENOMINAN SISTEMA OPERATIVO DE RED:
 - ◆ CADA MAQUINA TIENE UN ALTO GRADO DE AUTONOMIA.
 - ◆ EXISTEN POCOS REQUISITOS A LO LARGO DE TODO EL SISTEMA.

SISTEMAS OPERATIVOS DE REDES

- NFS PERMITE QUE CADA MAQUINA SEA UN CLIENTE Y UN SERVIDOR AL MISMO TIEMPO.
- CADA SERVIDOR DE NFS EXPORTA UNO O VARIOS DE SUS DIRECTORIOS (Y SUBDIRECTORIOS DEPENDIENTES) PARA EL ACCESO POR PARTE DE CLIENTES REMOTOS.
- LOS CLIENTES TIENEN ACCESO A LOS DIRECTORIOS EXPORTADOS MEDIANTE EL MONTAJE:
 - ◆ CUANDO UN CLIENTE MONTA UN DIRECTORIO (REMOTO), ESTE SE CONVIERTE EN PARTE DE SU JERARQUIA DE DIRECTORIOS.

SISTEMAS OPERATIVOS DE REDES

- NFS: NETWORK FILE SYSTEM.
- ES UNO DE LOS MAS CONOCIDOS Y ACEPTADO COMO SISTEMA OPERATIVO DE RED.
- FUE UN DESARROLLO DE SUN MICROSYSTEMS, SOPORTADO TAMBIEN POR DISTINTOS FABRICANTES:
 - ◆ SURGIO PARA UNIX PERO SE AMPLIO A OTROS S. O. (EJ.: MS - DOS).
 - ◆ SOPORTA SISTEMAS HETEROGENEOS:
 - EJ.: CLIENTES DE MS - DOS QUE HAGAN USO DE SERVIDORES UNIX.
 - ◆ LOS EQUIPOS PUEDEN SER TAMBIEN DE HARDWARE HETEROGENEO.

SISTEMAS OPERATIVOS DE REDES

- UN CLIENTE SIN DISCO PUEDE MONTAR UN ARCHIVO REMOTO EN SU DIRECTORIO RAIZ:
 - ◆ ESTO PRODUCE UN SISTEMA DE ARCHIVOS SOPORTADO EN SU TOTALIDAD EN UN SERVIDOR REMOTO.
- LAS ESTACIONES DE TRABAJO QUE NO POSEEN DISCOS LOCALES PUEDEN MONTAR DIRECTORIOS REMOTOS EN DONDE LO DESEEN, EN LA PARTE SUPERIOR DE SU JERARQUIA DE DIRECTORIOS LOCAL:
 - ◆ ESTA PRODUCE UN SISTEMA DE ARCHIVOS QUE ES EN PARTE LOCAL Y EN PARTE REMOTO.
- SI DOS O MAS CLIENTES MONTAN EL MISMO DIRECTORIO AL MISMO TIEMPO:
 - ◆ SE PUEDEN COMUNICAR AL COMPARTIR ARCHIVOS EN SUS DIRECTORIOS COMUNES.
 - ◆ NO HAY QUE HACER NADA ESPECIAL PARA LOGRAR COMPARTIR LOS ARCHIVOS.

SISTEMAS OPERATIVOS DE REDES

- LOS ARCHIVOS COMPARTIDOS FIGURAN EN LA JERARQUIA DE DIRECTORIOS DE VARIAS MAQUINAS:
 - ◆ SE LOS PUEDE LEER O ESCRIBIR DE LA MANERA USUAL.
- **PROTOCOLOS DE NFS.**
- UNO DE LOS OBJETIVOS DE NFS ES:
 - ◆ SOPORTAR UN SISTEMA HETEROGNEO EN DONDE LOS CLIENTES Y SERVIDORES PODRIAN EJECUTAR DISTINTOS S. O. EN HARDWARE DIVERSO:
 - ES ESENCIAL QUE LA INTERFAZ ENTRE LOS CLIENTES Y LOS SERVIDORES ESTE BIEN DEFINIDA.

SISTEMAS OPERATIVOS DE REDES

- SI EL NOMBRE DE LA RUTA DE ACCESO ES VALIDO Y EL DIRECTORIO ESPECIFICADO HA SIDO EXPORTADO:
 - ◆ EL SERVIDOR REGRESA UN ASA DE ARCHIVO (FILE HANDLE) AL CLIENTE:
 - CONTIENE CAMPOS QUE IDENTIFICAN:
 - DE MANERA UNICA EL TIPO DE SISTEMA DE ARCHIVOS, EL DISCO, EL N° DE NODO-I DEL DIRECTORIO.
 - LA INFORMACION RELATIVA A LA SEGURIDAD.
 - ES UTILIZADA EN LLAMADAS POSTERIORES PARA LA LECTURA O ESCRITURA DE ARCHIVOS EN EL DIRECTORIO MONTADO.

SISTEMAS OPERATIVOS DE REDES

- NFS LOGRA ESTE OBJETIVO DEFINIENDO DOS PROTOCOLOS CLIENTE - SERVIDOR:
 - ◆ UN PROTOCOLO ES UN CONJUNTO DE:
 - SOLICITUDES QUE ENVIAN LOS CLIENTES A LOS SERVIDORES.
 - LAS RESPUESTAS QUE ENVIAN LOS SERVIDORES DE REGRESO A LOS CLIENTES.

SISTEMAS OPERATIVOS DE REDES

- ALGUNOS S. O. SOPORTAN LA ALTERNATIVA DEL AUTOMONTAJE:
 - ◆ PERMITE QUE UN CONJUNTO DE DIRECTORIOS REMOTOS QUEDE ASOCIADO CON UN DIRECTORIO LOCAL.
 - ◆ NINGUNO DE LOS DIRECTORIOS REMOTOS SE MONTA DURANTE EL ARRANQUE DEL CLIENTE.
 - ◆ LA PRIMERA VEZ QUE SE ABRA UN ARCHIVO REMOTO, EL S. O. ENVIA UN MENSAJE A LOS SERVIDORES:
 - LOS SERVIDORES RESPONDEN Y SE MONTA SU DIRECTORIO.

SISTEMAS OPERATIVOS DE REDES

- UN PROTOCOLO DE NFS MANEJA EL MONTAJE.
- UN CLIENTE PUEDE:
 - ◆ ENVIAR EL NOMBRE DE UNA RUTA DE ACCESO A UN SERVIDOR.
 - ◆ SOLICITAR EL PERMISO PARA MONTAR ESE DIRECTORIO EN ALGUNA PARTE DE SU JERARQUIA DE DIRECTORIOS.

SISTEMAS OPERATIVOS DE REDES

- ◆ LAS PRINCIPALES VENTAJAS SOBRE EL MONTAJE ESTATICO SON:
 - SE EVITA EL TRABAJO DE CONTACTAR SERVIDORES Y MONTAR DIRECTORIOS QUE NO SON REQUERIDOS DE INMEDIATO.
 - SI EL CLIENTE PUEDE UTILIZAR VARIOS SERVIDORES EN PARALELO, SE PUEDE TENER:
 - CIERTA TOLERANCIA A FALLAS.
 - MEJORAR EL RENDIMIENTO.

SISTEMAS OPERATIVOS DE REDES

- NFS NO DA SOPORTE A LA DUPLICACION DE ARCHIVOS O DIRECTORIOS.
- OTRO PROTOCOLO DE NFS ES PARA EL ACCESO A LOS DIRECTORIOS Y ARCHIVOS.
- LOS CLIENTES PUEDEN:
 - ◆ ENVIAR MENSAJES A LOS SERVIDORES PARA EL MANEJO DE LOS DIRECTORIOS Y LA LECTURA O ESCRITURA DE ARCHIVOS.
 - ◆ TENER ACCESO A LOS ATRIBUTOS DE ARCHIVO, TALES COMO SU MODO, TAMAÑO Y FECHA DE LA ULTIMA MODIFICACION.
- NFS SOPORTA SERVIDORES SIN ESTADO:
 - ◆ NO MANTIENEN LA INFORMACION DE ESTADO RELATIVA A LOS ARCHIVOS ABIERTOS.
 - ◆ SI UN SERVIDOR FALLA Y ARRANCA RAPIDAMENTE, NO SE PIERDE INFORMACION ACERCA DE LOS ARCHIVOS ABIERTOS:
 - LOS PROGRAMAS CLIENTE NO FALLAN.

INTRODUCCION A LOS SISTEMAS DISTRIBUIDOS

85

SISTEMAS OPERATIVOS DE REDES

- LAS CLAVES UTILIZADAS PARA LA AUTENTIFICACION, ASI COMO OTRA INFORMACION, ESTAN CONTENIDAS EN EL NIS:
 - ◆ NETWORK INFORMATION SERVICE: SERVICIO DE INFORMACION DE LA RED.
 - ◆ ALMACENA PAREJAS (CLAVE, VALOR).
 - ◆ CUANDO SE PROPORCIONA UNA CLAVE, REGRESA EL VALOR CORRESPONDIENTE.
 - ◆ ALMACENA LA ASOCIACION DE:
 - LOS NOMBRES DE LOS USUARIOS CON LAS CONTRASEÑAS (CIFRADAS).
 - LOS NOMBRES DE LAS MAQUINAS CON LAS DIRECCIONES EN LA RED Y OTROS ELEMENTOS.

INTRODUCCION A LOS SISTEMAS DISTRIBUIDOS

88

SISTEMAS OPERATIVOS DE REDES

- EL SISTEMA DE ARCHIVOS REMOTOS (RFS) DEL SISTEMA V DE UNIX NO FUNCIONA ASI, SINO QUE:
 - ◆ EL SERVIDOR LLEVA UN REGISTRO DEL HECHO QUE CIERTO ARCHIVO ESTA ABIERTO Y LA POSICION ACTUAL DEL LECTOR.
 - ◆ SI UN SERVIDOR FALLA Y VUELVE A ARRANCAR RAPIDAMENTE:
 - SE PIERDEN TODAS LAS CONEXIONES ABIERTAS.
 - LOS PROGRAMAS CLIENTE FALLAN.
- EN UN SERVIDOR SIN ESTADO, COMO NFS:
 - ◆ LOS BLOQUEOS NO TIENEN QUE ASOCIARSE CON LOS ARCHIVOS ABIERTOS:
 - EL SERVIDOR NO SABE CUALES ARCHIVOS ESTAN ABIERTOS.
 - ◆ SE NECESITA UN MECANISMO ADICIONAL INDEPENDIENTE PARA CONTROLAR EL BLOQUEO.

INTRODUCCION A LOS SISTEMAS DISTRIBUIDOS

86

SISTEMAS OPERATIVOS DE REDES

- *IMPLANTACION DE NFS.*
- LA IMPLANTACION DEL CODIGO DEL CLIENTE Y EL SERVIDOR ES INDEPENDIENTE DE LOS PROTOCOLOS NFS.
- UNA IMPLEMENTACION QUE SUELE TOMARSE COMO REFERENCIA ES LA DE SUN, QUE CONSTA DE TRES CAPAS.
- LA CAPA SUPERIOR ES LA DE LLAMADAS AL SISTEMA:
 - ◆ MANEJA LAS LLAMADAS DEL TIPO OPEN, READ Y CLOSE.
 - ◆ ANALIZA LA LLAMADA Y VERIFICA LOS PARAMETROS.
 - ◆ LLAMA A LA SEGUNDA CAPA: CAPA DEL SISTEMA VIRTUAL DE ARCHIVOS: VIRTUAL FILE SYSTEM: VFS.
- LA CAPA VFS MANTIENE UNA TABLA CON UNA ENTRADA POR CADA ARCHIVO ABIERTO:
 - ◆ ES ANALOGA A LA TABLA DE NODOS-I PARA LOS ARCHIVOS ABIERTOS EN UNIX.

INTRODUCCION A LOS SISTEMAS DISTRIBUIDOS

89

SISTEMAS OPERATIVOS DE REDES

- NFS UTILIZA EL ESQUEMA DE PROTECCION DE UNIX, CON LOS BITS *rwex* PARA EL PROPIETARIO, GRUPO Y OTROS.
- SE PUEDE UTILIZAR LA CRIPTOGRAFIA DE CLAVES PUBLICAS PARA DAR VALIDEZ AL CLIENTE Y EL SERVIDOR EN CADA SOLICITUD Y RESPUESTA:
 - ◆ EL CLIENTE MALICIOSO NO PUEDE PERSONIFICAR A OTRO CLIENTE, YA QUE NO CONOCE SU CLAVE SECRETA.

INTRODUCCION A LOS SISTEMAS DISTRIBUIDOS

87

SISTEMAS OPERATIVOS DE REDES

- LA CAPA VFS TIENE UNA ENTRADA POR CADA ARCHIVO ABIERTO:
 - ◆ SE LA LLAMA NODO-V (NODO-I VIRTUAL).
 - ◆ LOS NODOS-V SE UTILIZAN PARA INDICAR SI EL ARCHIVO ES LOCAL O REMOTO.
 - ◆ PARA LOS ARCHIVOS REMOTOS, POSEEN LA INFORMACION SUFICIENTE COMO PARA TENER ACCESO A ELLOS.

INTRODUCCION A LOS SISTEMAS DISTRIBUIDOS

90

SISTEMAS OPERATIVOS DE REDES

- PARA MONTAR UN SISTEMA REMOTO DE ARCHIVOS, EL ADMINISTRADOR DEL SISTEMA LLAMA AL PROGRAMA *MOUNT* :
 - ◆ UTILIZA LA INFORMACION DEL DIRECTORIO REMOTO, EL DIRECTORIO LOCAL DONDE SERA MONTADO Y OTROS DATOS ADICIONALES.
 - ◆ CON EL NOMBRE DEL DIRECTORIO REMOTO POR MONTAR SE DESCUBRE EL NOMBRE DE LA MAQUINA DONDE SE LOCALIZA DICHO DIRECTORIO.
 - ◆ SE VERIFICA SI EL DIRECTORIO EXISTE Y SI ESTA DISPONIBLE PARA SU MONTAJE REMOTO.

SISTEMAS OPERATIVOS DE REDES

- UN CRITERIO SIMILAR SE SIGUE CON LA ESCRITURA:
 - ◆ ANTES DE SER ENVIADOS AL SERVIDOR LOS DATOS SE ACUMULAN EN FORMA LOCAL:
 - HASTA COMPLETAR CIERTA CANTIDAD DE BYTES.
 - HASTA QUE SE CIERRA EL ARCHIVO.
- OTRA TECNICA UTILIZADA PARA MEJORAR EL RENDIMIENTO ES EL OCULTAMIENTO O CACHING:
 - ◆ LOS SERVIDORES OCULTAN LOS DATOS PARA EVITAR EL ACCESO AL DISCO:
 - ESTO ES INVISIBLE PARA LOS CLIENTES.

SISTEMAS OPERATIVOS DE REDES

- EL NUCLEO:
 - ◆ CONSTRUYE UN NODO-V PARA EL DIRECTORIO REMOTO.
 - ◆ PIDE EL CODIGO DEL CLIENTE NFS PARA CREAR UN NODO-R (NODO-I REMOTO) EN SUS TABLAS INTERNAS.
- EL NODO-V APUNTA AL NODO-R.
- CADA NODO-V DE LA CAPA VFS CONTENDRA EN ULTIMA INSTANCIA UN APUNTADOR A UN NODO-I EN EL S. O. LOCAL.
- ES POSIBLE VER DESDE EL NODO-V SI UN ARCHIVO O DIRECTORIO ES LOCAL O REMOTO Y, SI ES REMOTO, ENCONTRAR SU ASA DE ARCHIVO.
- TODO ARCHIVO O DIRECTORIO ABIERTO TIENE UN NODO-V QUE APUNTA A UN NODO-R O A UN NODO-I.

SISTEMAS OPERATIVOS DE REDES

- LOS CLIENTES MANTIENEN DOS CACHES:
 - ◆ UNO PARA LOS ATRIBUTOS DE ARCHIVO (NODOS-I).
 - ◆ OTRO PARA LOS DATOS DEL ARCHIVO.
- CUANDO SE NECESITA UN NODO-I O UN BLOQUE DEL ARCHIVO:
 - ◆ PRIMERO SE VERIFICA SI LA SOLICITUD SE PUEDE SATISFACER MEDIANTE EL CACHE DEL CLIENTE:
 - SE EVITA EL TRAFICO EN LA RED.

SISTEMAS OPERATIVOS DE REDES

- POR RAZONES DE EFICIENCIA LAS TRANSFERENCIAS ENTRE CLIENTE Y SERVIDOR SE HACEN EN BLOQUES GRANDES, GENERALMENTE DE 8K:
 - ◆ LUEGO DE HABER RECIBIDO LA CAPA VFS DEL CLIENTE EL BLOQUE NECESARIO, EMITE LA SOLICITUD DEL SIGUIENTE BLOQUE:
 - ESTO SE DENOMINA LECTURA ADELANTADA (READ AHEAD).

SISTEMAS OPERATIVOS DE REDES

- UN PROBLEMA IMPORTANTE DEL CACHING ES QUE EL CACHE NO ES COHERENTE, EJ.:
 - ◆ DOS CLIENTES OCULTAN EL MISMO BLOQUE DEL ARCHIVO.
 - ◆ UNO DE ELLOS LO MODIFICA.
 - ◆ CUANDO EL OTRO LEE EL BLOQUE, OBTIENE EL VALOR ANTIGUO.

SISTEMAS OPERATIVOS DE REDES

- ◆ PARA MITIGAR ESTE PROBLEMA, LA IMPLANTACION DE NFS:
 - ASOCIA A CADA BLOQUE CACHE UN TEMPORIZADOR (TIMER).
 - CUANDO EL TIMER EXPIRA, LA ENTRADA SE DESCARTA.
 - GENERALMENTE LOS TIEMPOS SON DE:
 - 3 SEG. PARA BLOQUES DE DATOS.
 - 30 SEG. PARA BLOQUES DE DIRECTORIO.
- ◆ AL ABRIR UN ARCHIVO CON CACHE:
 - SE ENVIA UN MENSAJE AL SERVIDOR PARA REVISAR LA HORA DE LA ULTIMA MODIFICACION.
- ◆ SE DETERMINA SI LA COPIA DEL CACHE ES VALIDA O DEBE DESCARTARSE, UTILIZANDO UNA NUEVA COPIA DEL SERVIDOR.
- ◆ EL TEMPORIZADOR DEL CACHE EXPIRA CADA 30 SEG.:
 - TODOS LOS BLOQUES MODIFICADOS EN EL CACHE SE ENVIAN AL SERVIDOR.

SISTEMAS REALMENTE DISTRIBUIDOS

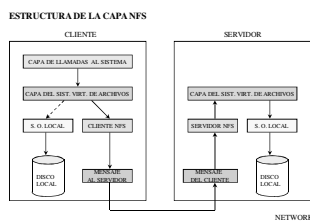
SISTEMAS OPERATIVOS DE REDES

- RESUMIENDO:
 - ◆ NFS SOLO TRATA EL SISTEMA DE ARCHIVOS.
 - ◆ NFS NO HACE REFERENCIA A OTROS ASPECTOS, COMO LA EJECUCION DE UN PROCESO.
 - ◆ NFS SE HA DIFUNDIDO AMPLIAMENTE, A PESAR DE TODO.

SISTEMAS REALMENTE DISTRIBUIDOS

- NFS ES UN EJEMPLO DE SOFTWARE DEBILMENTE ACOPLADO EN HARDWARE DEBILMENTE ACOPLADO:
 - ◆ CADA COMPUTADORA PUEDE EJECUTAR SU PROPIO S. O.
 - ◆ SOLO SE DISPONE DE UN SISTEMA COMPARTIDO DE ARCHIVOS.
 - ◆ EL TRAFICO CLIENTE - SERVIDOR DEBE OBEDECER LOS PROTOCOLOS NFS.
- LAS MULTICOMPUTADORAS SON UN EJEMPLO DE SOFTWARE FUERTEMENTE ACOPLADO EN HARDWARE DEBILMENTE ACOPLADO:
 - ◆ CREAN LA ILUSION DE QUE TODA LA RED DE COMPUTADORAS ES UN SOLO SISTEMA DE TIEMPO COMPARTIDO, EN VEZ DE UNA COLECCION DE MAQUINAS DIVERSAS.

SISTEMAS OPERATIVOS DE REDES



SISTEMAS REALMENTE DISTRIBUIDOS

- UN SISTEMA DISTRIBUIDO ES AQUEL QUE SE EJECUTA EN UNA COLECCION DE MAQUINAS SIN MEMORIA COMPARTIDA, PERO QUE APARECE ANTE SUS USUARIOS COMO UNA SOLA COMPUTADORA:
 - ◆ A ESTA PROPIEDAD SE LA CONOCE COMO LA IMAGEN DE UN UNICO SISTEMA.
- TAMBIEN SE DEFINE UN SISTEMA DISTRIBUIDO COMO AQUEL QUE SE EJECUTA EN UNA COLECCION DE MAQUINAS ENLAZADAS MEDIANTE UNA RED PERO QUE ACTUAN COMO UN UNIPROCESADOR VIRTUAL.

SISTEMAS REALMENTE DISTRIBUIDOS

- ALGUNAS DE LAS CARACTERISTICAS DE LOS SISTEMAS DISTRIBUIDOS SON LAS SIGUIENTES:
 - ◆ DEBE EXISTIR UN MECANISMO DE COMUNICACION GLOBAL ENTRE LOS PROCESOS:
 - CUALQUIER PROCESO DEBE PODER COMUNICARSE (INTERCAMBIAR INFORMACION) CON CUALQUIER OTRO.
 - ◆ NO TIENE QUE HABER:
 - DISTINTOS MECANISMOS EN DISTINTAS MAQUINAS.
 - DISTINTOS MECANISMOS PARA LA COMUNICACION LOCAL O LA COMUNICACION REMOTA.

SISTEMAS DE MULTIPROCESADOR CON TIEMPO COMPARTIDO

- CORRESPONDE A SOFTWARE FUERTEMENTE ACOPLADO EN HARDWARE FUERTEMENTE ACOPLADO.
- LOS EJEMPLOS MAS COMUNES DE PROPOSITO GENERAL SON LOS MULTIPROCESADORES:
 - ◆ OPERAN COMO UN SISTEMA DE TIEMPO COMPARTIDO, PERO CON VARIAS CPU EN VEZ DE UNA SOLA.
 - ◆ EXTERNAMENTE UN MULTIPROCESADOR CON 32 CPU DE 3 MIPS ACTUA DE MANERA MUY PARECIDA A UNA SOLA CPU DE 96 MIPS:
 - 1 MIPS: 1.000.000 DE INSTRUCCIONES POR SEGUNDO.
 - ◆ SE CORRESPONDE CON LA IMAGEN DE UN UNICO SISTEMA.

SISTEMAS REALMENTE DISTRIBUIDOS

- ◆ DEBE EXISTIR UN ESQUEMA GLOBAL DE PROTECCION.
- ◆ LA ADMINISTRACION DE PROCESOS DEBE SER LA MISMA EN TODAS PARTE.
- ◆ SE DEBE TENER UNA MISMA INTERFAZ DE LLAMADAS AL SISTEMA EN TODAS PARTES:
 - ES NORMAL QUE SE EJECUTEN NUCLEOS IDENTICOS EN TODAS LAS CPU DEL SISTEMA.
- ◆ ES NECESARIO UN SISTEMA GLOBAL DE ARCHIVOS.

SISTEMAS DE MULTIPROCESADOR CON TIEMPO COMPARTIDO

- LA CARACTERISTICA CLAVE ES LA EXISTENCIA DE UNA SOLA COLA PARA EJECUCION:
 - ◆ UNA LISTA DE TODOS LOS PROCESOS EN EL SISTEMA QUE NO ESTAN BLOQUEADOS EN FORMA LOGICA Y LISTOS PARA SU EJECUCION.
 - ◆ LA COLA DE EJECUCION ES UNA ESTRUCTURA DE DATOS CONTENIDA EN LA MEMORIA COMPARTIDA.
- LOS PROGRAMAS DE LOS PROCESOS ESTAN EN LA MEMORIA COMPARTIDA, TAMBIEN EL S. O.
- EL PLANIFICADOR (DE PROCESOS) DEL S. O. SE EJECUTA COMO UNA "REGION CRITICA":
 - ◆ SE EVITA QUE DOS CPU ELIJAN EL MISMO PROCESO PARA SU EJECUCION INMEDIATA.

SISTEMAS DE MULTIPROCESADOR CON TIEMPO COMPARTIDO

SISTEMAS DE MULTIPROCESADOR CON TIEMPO COMPARTIDO

- CUANDO UN PROCESO SE ASIGNA A UN PROCESADOR:
 - ◆ ENCUENTRA QUE EL CACHE DEL PROCESADOR ESTA OCUPADO POR PALABRAS DE MEMORIA QUE PERTENECEN A AQUELLA PARTE DE LA MEMORIA COMPARTIDA QUE CONTIENE AL PROGRAMA DEL PROCESO ANTERIOR.
 - ◆ LUEGO DE UN BREVE LAPSO SE HABRAN REEMPLAZADO POR EL CODIGO Y LOS DATOS DEL PROGRAMA DEL PROCESO ASIGNADO A ESE PROCESADOR.
- NINGUNA CPU TIENE MEMORIA LOCAL:
 - ◆ TODOS LOS PROGRAMAS SE ALMACENAN EN LA MEMORIA GLOBAL COMPARTIDA.

**SISTEMAS DE MULTIPROCESADOR
CON TIEMPO COMPARTIDO**

- SI TODAS LAS CPU ESTAN INACTIVAS EN ESPERA DE E / S Y UN PROCESO ESTA LISTO PARA SU EJECUCION:
 - ◆ ES CONVENIENTE ASIGNARLO A LA CPU QUE SE UTILIZO POR ULTIMA VEZ (PARA ESE PROCESO):
 - LA HIPOTESIS ES QUE NINGUN OTRO PROCESO UTILIZO ESA CPU DESDE ENTONCES (HIPOTESIS DE VASWANI Y ZAHORJAN).

ASPECTOS DEL DISEÑO

**SISTEMAS DE MULTIPROCESADOR
CON TIEMPO COMPARTIDO**

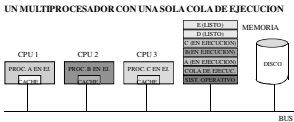
- SI UN PROCESO SE BLOQUEA EN ESPERA DE E / S EN UN MULTIPROCESADOR, EL S. O. PUEDE:
 - ◆ SUSPENDERLO.
 - ◆ DEJARLO EN "ESPERA OCUPADA":
 - ES APLICABLE CUANDO LA MAYORIA DE LA E / S SE REALIZA EN MENOS TIEMPO DEL QUE TARDA UN CAMBIO ENTRE LOS PROCESOS.
 - EL PROCESO CONSERVA SU PROCESADOR POR ALGUNOS MILLISEGUNDOS EN ESPERA DE QUE LA E / S FINALICE:
 - SI SE AGOTA EL TIEMPO DE ESPERA Y NO HA FINALIZADO LA E / S, SE REALIZA UNA CONMUTACION DE PROCESOS.

ASPECTOS DEL DISEÑO

- LA COMPARACION DE LAS TRES PRINCIPALES FORMAS DE ORGANIZAR N CPU SE PUEDE RESUMIR EN EL SIGUIENTE CUADRO:

**SISTEMAS DE MULTIPROCESADOR
CON TIEMPO COMPARTIDO**

- GENERALMENTE SE DISPONDRA DE UN SISTEMA DE ARCHIVOS TRADICIONAL, CON UN UNICO CACHE:
 - ◆ GLOBALMENTE CONSIDERADO ES SIMILAR AL SISTEMA DE ARCHIVOS DE UN UNICO PROCESADOR.



ASPECTOS DEL DISEÑO

ELEMENTO	S. O. DE RED	S. O. DISTRIB.	S. O. DE MULTIPR.
¿SE VE COMO UN UNIPROCESADOR VIRTUAL?	NO	SI	SI
¿TODAS TIENEN QUE EJECUTAR EL MISMO S. O.?	NO	SI	SI
¿CUANTAS COPIAS DEL S. O. EXISTEN?	N	N	1
¿COMO SE LOGRA LA COMUNICACION?	ARCHIVOS COMPARTIDOS	MENSAJES	MEMORIA COMPARTIDA
¿SE REQUIERE UN ACUERDO EN LOS PROTOCOLOS DE LA RED?	SI	SI	NO
¿EXISTE UNA UNICA COLA DE EJECUCION?	NO	NO	SI
¿EXISTE UNA SEMANTICA BIEN DEFINIDA PARA LOS ARCHIVOS COMPARTIDOS?	POR LO GENERAL NO	SI	SI

ASPECTOS DEL DISEÑO

- LOS ASPECTOS CLAVES EN EL DISEÑO DE S. O. DISTRIBUIDOS SON:
 - ◆ TRANSPARENCIA, FLEXIBILIDAD, CONFIABILIDAD, DESEMPEÑO Y ESCALABILIDAD.

TRANSPARENCIA

- LA TRANSPARENCIA *DESDE EL PUNTO DE VISTA DE LOS PROGRAMAS* SIGNIFICA DISEÑAR LA INTERFAZ DE LLAMADAS AL SISTEMA DE MODO QUE NO SEA VISIBLE LA EXISTENCIA DE VARIOS PROCESADORES.
- NO ES TRANSPARENTE UN SISTEMA DONDE EL ACCESO A LOS ARCHIVOS REMOTOS SE REALICE MEDIANTE:
 - ◆ EL ESTABLECIMIENTO EXPLICITO DE UNA CONEXION EN LA RED CON UN SERVIDOR REMOTO.
 - ◆ EL ENVIO POSTERIOR DE MENSAJES:
 - EL ACCESO A LOS SERVICIOS REMOTOS SERA DISTINTO AL ACCESO A LOS SERVICIOS LOCALES.

TRANSPARENCIA

TRANSPARENCIA

- EXISTEN DISTINTOS TIPOS DE TRANSPARENCIA EN UN SISTEMA DISTRIBUIDO:
 - ◆ DE LOCALIZACION: LOS USUARIOS NO PUEDEN INDICAR LA LOCALIZACION DE LOS RECURSOS.
 - ◆ DE MIGRACION: LOS RECURSOS SE PUEDEN MOVER A VOLUNTAD SIN CAMBIAR SUS NOMBRES.
 - ◆ DE REPLICA: LOS USUARIOS NO PUEDEN INDICAR EL N° DE COPIAS EXISTENTES.
 - ◆ DE CONCURRENCIA: VARIOS USUARIOS PUEDEN COMPARTIR RECURSOS DE MANERA AUTOMATICA.
 - ◆ DE PARALELISMO: LAS ACTIVIDADES PUEDEN OCURRIR EN PARALELO SIN EL CONOCIMIENTO DE LOS USUARIOS.

TRANSPARENCIA

- UN ASPECTO MUY IMPORTANTE ES LA FORMA DE LOGRAR LA IMAGEN DE UN UNICO SISTEMA.
- LOS USUARIOS DEBEN PERCIBIR QUE LA COLECCION DE MAQUINAS CONECTADAS SON UN SISTEMA DE TIEMPO COMPARTIDO DE UN SOLO PROCESADOR:
 - ◆ UN SISTEMA QUE LOGRE ESTE OBJETIVO SE DICE QUE ES TRANSPARENTE.
- *DESDE EL PUNTO DE VISTA DE LOS USUARIOS*, LA TRANSPARENCIA SE LOGRA CUANDO:
 - ◆ SUS PEDIDOS SE SATISFACEN CON EJECUCIONES EN PARALELO EN DISTINTAS MAQUINAS.
 - ◆ SE UTILIZAN UNA VARIEDAD DE SERVIDORES DE ARCHIVOS.
 - ◆ EL USUARIO NO NECESITA SABERLO NI NOTARLO.

FLEXIBILIDAD

FLEXIBILIDAD

- LA FLEXIBILIDAD ES DE FUNDAMENTAL IMPORTANCIA.
- EXISTEN DOS ESCUELAS DE PENSAMIENTO EN CUANTO A LA ESTRUCTURA DE LOS SISTEMAS DISTRIBUIDOS:
 - ◆ NUCLEO MONOLITICO:
 - CADA MAQUINA DEBE EJECUTAR UN NUCLEO TRADICIONAL QUE PROPORCIONE LA MAYORIA DE LOS SERVICIOS.
 - ◆ MICRONUCLEO (MICROKERNEL):
 - EL NUCLEO DEBE PROPORCIONAR LO MENOS POSIBLE.
 - EL GRUESO DE LOS SERVICIOS DEL S. O. SE DEBE OBTENER A PARTIR DE LOS SERVIDORES AL NIVEL USUARIO.

FLEXIBILIDAD

- EL OBJETIVO ES MANTENER EL MICRONUCLEO PEQUEÑO.
- TODOS LOS DEMAS SERVICIOS DEL S. O. SE IMPLEMENTAN GENERALMENTE COMO SERVIDORES A NIVEL USUARIO:
 - ◆ PARA OBTENER UN SERVICIO:
 - EL USUARIO ENVIA UN MENSAJE AL SERVIDOR APROPIADO.
 - EL SERVIDOR REALIZA EL TRABAJO Y REGRESA EL RESULTADO.

FLEXIBILIDAD

- EL NUCLEO MONOLITICO ES EL S. O. CENTRALIZADO AUMENTADO CON:
 - ◆ CAPACIDADES DE RED.
 - ◆ INTEGRACION DE SERVICIOS REMOTOS.
- CON NUCLEO MONOLITICO:
 - ◆ LA MAYORIA DE LAS LLAMADAS AL SISTEMA SE REALIZAN MEDIANTE SEÑALAMIENTO AL NUCLEO:
 - EL NUCLEO REALIZA EL TRABAJO.
 - EL NUCLEO REGRESA EL RESULTADO AL PROCESO DEL USUARIO.
 - ◆ LA MAYORIA DE LAS MAQUINAS TIENE DISCOS Y ADMINISTRA SUS PROPIOS SISTEMAS LOCALES DE ARCHIVOS.

FLEXIBILIDAD

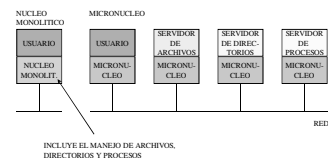
- UNA IMPORTANTE VENTAJA DE ESTE METODO ES SU ALTA MODULARIDAD:
 - ◆ EXISTE UNA INTERFAZ BIEN DEFINIDA CON CADA SERVICIO (CONJUNTO DE MENSAJES QUE COMPRENDE EL SERVIDOR).
 - ◆ CADA SERVICIO ES IGUAL DE ACCESIBLE PARA TODOS LOS CLIENTES, INDEPENDIENTEMENTE DE LA POSICION.
 - ◆ ES FACIL IMPLANTAR, INSTALAR Y DEPURAR NUEVOS SERVICIOS, SIN NECESIDAD DE DETENER EL SISTEMA TOTALMENTE.

FLEXIBILIDAD

- EL MICRONUCLEO ES MAS FLEXIBLE Y PROPORCIONA SOLO CUATRO SERVICIOS MINIMOS:
 - ◆ UN MECANISMO DE COMUNICACION ENTRE PROCESOS.
 - ◆ CIERTA ADMINISTRACION DE LA MEMORIA.
 - ◆ UNA CANTIDAD LIMITADA DE PLANIFICACION Y ADMINISTRACION DE PROCESOS DE BAJO NIVEL.
 - ◆ ENTRADA / SALIDA DE BAJO NIVEL.
- CONTRARIAMENTE AL NUCLEO MONOLITICO, EL MICRONUCLEO NO PROPORCIONA:
 - ◆ EL SISTEMA DE ARCHIVOS, EL SISTEMA DE DIRECTORIOS, TODA LA ADMINISTRACION DE PROCESOS O GRAN PARTE DEL MANEJO DE LAS LLAMADAS AL SISTEMA.

FLEXIBILIDAD

ESQUEMA DE NUCLEO MONOLITICO Y MICRONUCLEO



CONFIABILIDAD

CONFIABILIDAD

- LOS DATOS NO DEBEN PERDERSE O MEZCLARSE:
 - ◆ SI LOS ARCHIVOS SE ALMACENAN DE MANERA REDUNDANTE EN VARIOS SERVIDORES, TODAS LAS COPIAS DEBEN SER CONSISTENTES.
- OTRO ASPECTO DE LA CONFIABILIDAD GENERAL ES LA SEGURIDAD:
 - ◆ LOS ARCHIVOS Y OTROS RECURSOS DEBEN SER PROTEGIDOS CONTRA EL USO NO AUTORIZADO.
- OTRO ASPECTO RELACIONADO CON LA CONFIABILIDAD ES LA TOLERANCIA A FALLAS:
 - ◆ LAS FALLAS SE DEBEN OCULTAR BRINDANDO UNA RECUPERACION TRANSPARENTE PARA EL USUARIO, AUNQUE HAYA CIERTA DEGRADACION DE LA PERFORMANCE.

CONFIABILIDAD

- UN IMPORTANTE OBJETIVO DE LOS SISTEMAS DISTRIBUIDOS ES QUE SI UNA MAQUINA FALLA, ALGUNA OTRA DEBE ENCARGARSE DEL TRABAJO.
- LA CONFIABILIDAD GLOBAL *TEORICA* DEL SISTEMA PODRIA SER EL “OR” BOOLEANO DE LA CONFIABILIDAD DE LOS COMPONENTES; EJEMPLO:
 - ◆ SE DISPONE DE 5 SERVIDORES DE ARCHIVOS, C / U CON UNA PROBABILIDAD DE 0,95 DE FUNCIONAR EN UN INSTANTE DADO.
 - ◆ LA PROBABILIDAD DE FALLA SIMULTANEA DE LOS 5 ES $(0,05)^5 = 0,000006$.
 - ◆ LA PROBABILIDAD DE QUE AL MENOS UNO ESTE DISPONIBLE ES 0,999994.

DESEMPEÑO

CONFIABILIDAD

- LA CONFIABILIDAD *PRACTICA* SE VE DISMINUIDA YA QUE MUCHAS VECES SE REQUIERE QUE CIERTOS SERVIDORES ESTEN EN SERVICIO SIMULTANEAMENTE PARA QUE EL TODO FUNCIONE:
 - ◆ ALGUNOS SISTEMAS TIENEN UNA DISPONIBILIDAD MAS RELACIONADA CON EL “AND” BOOLEANO DE LAS COMPONENTES QUE CON EL “OR” BOOLEANO.
- UN ASPECTO DE LA CONFIABILIDAD ES LA DISPONIBILIDAD:
 - ◆ SE REFIERE A LA FRACCION DE TIEMPO EN QUE SE PUEDE UTILIZAR EL SISTEMA.
- LA DISPONIBILIDAD SE MEJORA MEDIANTE:
 - ◆ UN DISEÑO QUE NO EXIJA EL FUNCIONAMIENTO SIMULTANEO DE UN N° SUSTANCIAL DE COMPONENTES CRITICOS.
 - ◆ LA REDUNDANCIA, ES DECIR LA DUPLICIDAD DE COMPONENTES CLAVE DEL HARDWARE Y DEL SOFTWARE.

DESEMPEÑO

- CUANDO SE EJECUTA UNA APLICACION EN UN SISTEMA DISTRIBUIDO NO DEBE PARECER PEOR QUE SU EJECUCION EN UN UNICO PROCESADOR:
 - ◆ ESTO ES DIFICIL DE LOGRAR.
- ALGUNAS METRICAS DEL DESEMPEÑO SON:
 - ◆ TIEMPO DE RESPUESTA.
 - ◆ RENDIMIENTO (N° DE TRABAJOS POR HORA).
 - ◆ USO DEL SISTEMA Y CANTIDAD CONSUMIDA DE LA CAPACIDAD DE LA RED.
- EL PROBLEMA SE COMPLICA POR EL HECHO DE QUE LA COMUNICACION ENTRE EQUIPOS ES LENTA COMPARADA CON:
 - ◆ LA VELOCIDAD DE PROCESO.
 - ◆ LA VELOCIDAD DE LA COMUNICACION DENTRO DE UN MISMO PROCESADOR.

DESEMPEÑO

- SE REQUIERE EL USO DE **PROTOCOLOS DE COMUNICACIONES** EN LOS EXTREMOS (PROCESADORES) QUE INTERVIENEN EN LA COMUNICACION:
 - ◆ SE INCREMENTA EL CONSUMO DE CICLOS DE PROCESADOR.
- PARA **OPTIMIZAR EL DESEMPEÑO** FRECUENTEMENTE HAY QUE:
 - ◆ **MINIMIZAR EL N° DE MENSAJES:**
 - LA DIFICULTAD ES QUE LA MEJOR FORMA DE MEJORAR EL DESEMPEÑO ES TENER MUCHAS ACTIVIDADES EN EJECUCION PARALELA EN **DISTINTOS PROCESADORES:**
 - ESTO **REQUIERE** EL ENVIO DE MUCHOS MENSAJES.
 - ◆ **CENTRALIZAR EL TRABAJO EN UNA SOLA MAQUINA:**
 - RESULTA **POCO APROPIADO** PARA UN SISTEMA **DISTRIBUIDO.**

ESCALABILIDAD

- LA **TENDENCIA** INDICA QUE EL TAMAÑO DE LOS SISTEMAS DISTRIBUIDOS ES HACIA **CIENTOS DE MILES Y AUN DECENAS DE MILLONES DE USUARIOS CONECTADOS.**
- EXISTEN **CUELLOS DE BOTELLA POTENCIALES** QUE SE DEBE INTENTAR EVITAR EN LOS SISTEMAS DISTRIBUIDOS DE GRAN ESCALA:
 - ◆ **COMPONENTES CENTRALIZADOS:**
 - EJ.: UN SOLO SERVIDOR DE CORREO PARA TODOS LOS USUARIOS.
 - ◆ **TABLAS CENTRALIZADAS:**
 - EJ.: UN UNICO DIRECTORIO TELEFONICO EN LINEA.
 - ◆ **ALGORITMOS CENTRALIZADOS:**
 - EJ.: REALIZACION DE UN RUTEO CON BASE EN LA INFORMACION COMPLETA.

DESEMPEÑO

- TAMBIEN SE DEBE PRESTAR ATENCION AL **TAMAÑO DE GRANO** DE TODOS LOS CALCULOS:
 - ◆ **PARALELISMO DE GRANO FINO:**
 - CORRESPONDE A TRABAJOS CON UN **GRAN N° DE PEQUEÑOS CALCULOS** Y MUCHA INTERACCION CON OTROS TRABAJOS:
 - REQUIEREN **MUCHA COMUNICACION** QUE PUEDE AFECTAR EL DESEMPEÑO.
 - ◆ **PARALELISMO DE GRANO GRUESO:**
 - CORRESPONDE A TRABAJOS CON **GRANDES CALCULOS, Poca INTERACCION Y POCOS DATOS:**
 - REQUIEREN **POCA COMUNICACION** Y NO AFECTAN LA PERFORMANCE.

ESCALABILIDAD

- SE DEBEN UTILIZAR **ALGORITMOS DESCENTRALIZADOS** CON LAS SIGUIENTES CARACTERISTICAS:
 - ◆ **NINGUNA MAQUINA TIENE LA INFORMACION COMPLETA** ACERCA DEL ESTADO DEL SISTEMA.
 - ◆ LAS **MAQUINAS TOMAN DECISIONES** SOLO EN BASE A LA **INFORMACION** DISPONIBLE DE MANERA **LOCAL.**
 - ◆ EL **FALLO DE UNA MAQUINA NO ARRUIÑA** EL ALGORITMO.
 - ◆ **NO EXISTE UNA HIPOTESIS IMPLICITA** DE LA EXISTENCIA DE UN **RELOJ GLOBAL.**

ESCALABILIDAD