



Introducción a los sistemas distribuidos

Roberto Gómez Cárdenas
<http://homepage.cem.itesm.mx/rogomez>
rogomez@itesm.mx

Lámina 1 Dr. Roberto Gómez Cárdenas



Hacer Con-ciencia

Alguien pidió una explicación de la teoría de Einstein. El profesor con mucho gusto le habló de tensores y geodésicas tetradimensionales.

- *No he entendido una sola palabra* - dijo el estudiante estupefacto.

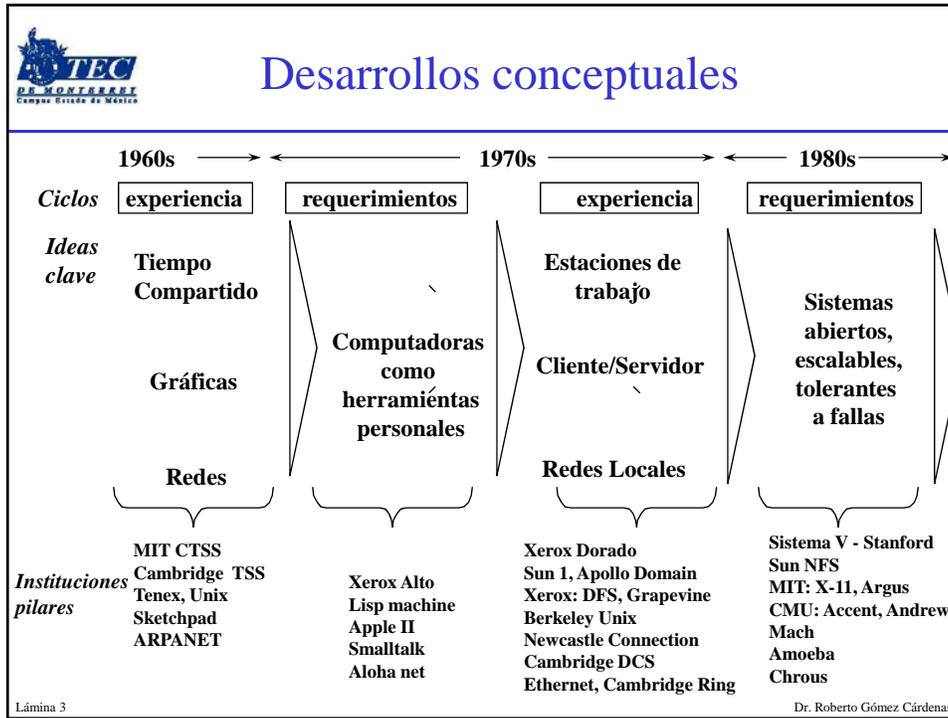
El profesor reflexionó un momento, y con menos entusiasmo le da una explicación menos técnica, conservando algunas geodésicas, pero haciendo intervenir aviadores y disparos de revolver.

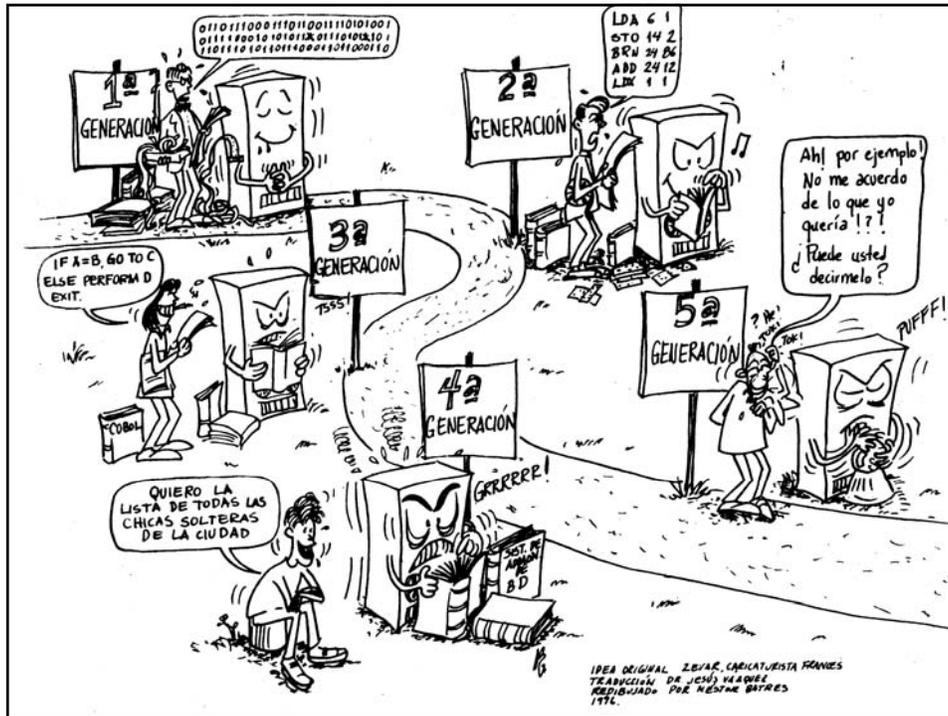
Ya entiendo casi todo - dijo el estudiante con bastante alegría - *pero hay algo que casi no entiendo, esas geodésicas, esas coordenadas ...*

De primido, el profesor piensa un largo rato y se dedica exclusivamente a aviadores que fuman mientras viajan a la velocidad de la luz, jefes de estación que disparan un revólver con la mano derecha y verifican tiempos en un cronómetro que tienen en la mano izquierda, trenes y campanas.

- *Ahora sí entiendo la relatividad* - exclamó el estudiante con alegría.
- *Sí* - le respondió amargamente el profesor, - *pero ahora ya NO ES la relatividad*

Lámina 2 ² Dr. Roberto Gómez Cárdenas







Antecedentes

- Dos grandes avances tecnológicos en los 80's
 - Desarrollo de microprocesadores
 - De una máquina de 10 millones de dólares que ejecuta una instrucción por minuto, se pasa a máquinas de 1000 dólares que ejecutan 10 millones de instrucciones por seg..
 - Invención de redes
 - Posibilidad de conectar uno o más computadoras entre sí
 - Pequeña observación:
 - Roll Royce de 100 dólares con un billón de kilómetros por litro
 - Tamaño manual para abrir puerta: 200 pgs.



Lámina 6

Dr. Roberto Gómez Cárdenas

 ¿Qué es un sistema distribuido?

Una colección de dispositivos independientes que es vista como un solo sistema independiente



Lámina 7 Dr. Roberto Gómez Cárdenas

 Características

- Recursos accesibles
- Transparencia
- Apertura
- Escalabilidad

Lámina 8 Dr. Roberto Gómez Cárdenas

 **Recursos computacionales**

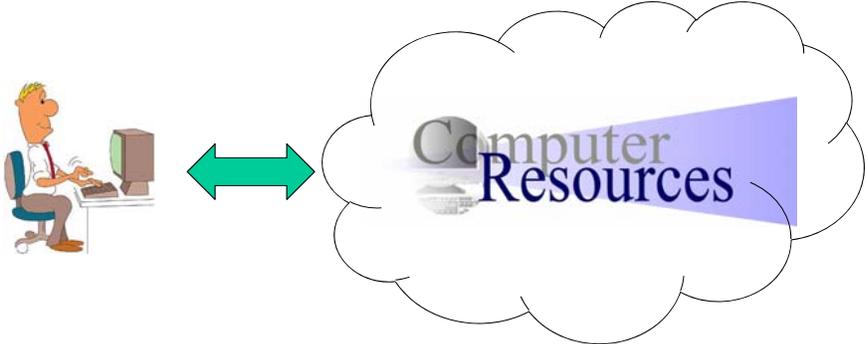


Lámina 9 Dr. Roberto Gómez Cárdenas

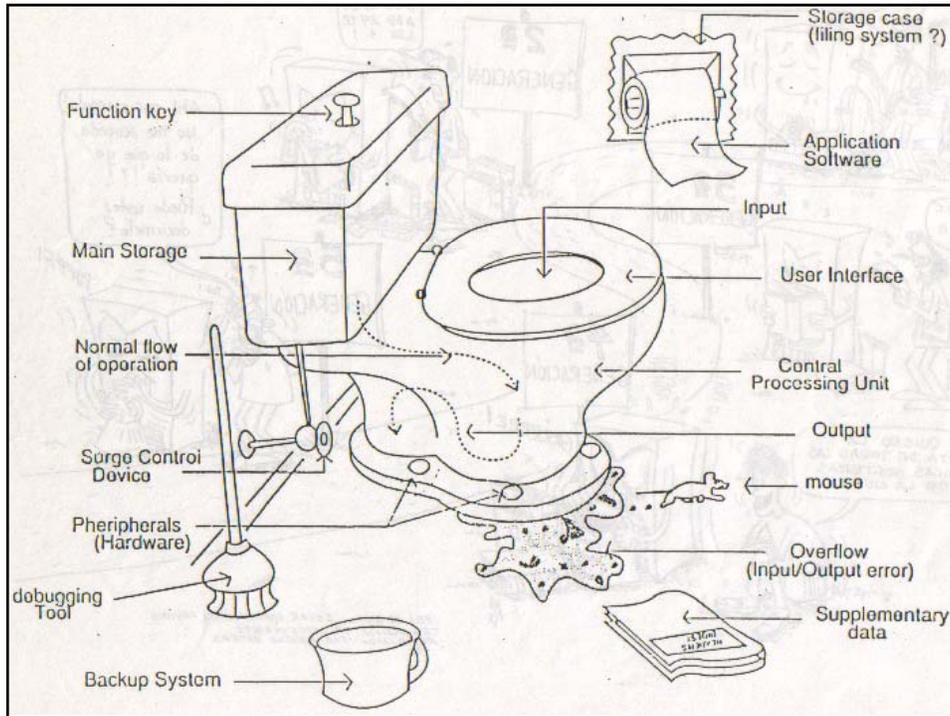
The diagram illustrates the concept of cloud computing. On the left, a cartoon character is seated at a desk with a computer. A green double-headed arrow connects this local workstation to a cloud on the right. Inside the cloud, the words 'Computer Resources' are written in a stylized font, with 'Computer' in grey and 'Resources' in blue. A purple rectangular shape is positioned behind the text, suggesting a layer of abstraction or a specific resource type.

 **Hay de recursos, a recursos**



Lámina 10 Dr. Roberto Gómez Cárdenas

The photograph shows a person's hands typing on a keyboard. The desk is extremely cluttered with various items, including a bottle of GEROL SPRING water, a pack of LUCKY STRIKE cigarettes, a pack of Pirella GARDOLINI cigarettes, a lit cigarette, a computer mouse, and numerous tangled cables. This visual metaphor suggests that the abundance of physical resources (like a messy desk) does not necessarily translate to efficient or organized digital resources.



 **TEC**
DE MONTERREY
Campus Estado de México

Transparencia



Esconder el hecho de que los procesos y los recursos se encuentran físicamente distribuidos entre varios dispositivos

Lámina 12 Dr. Roberto Gómez Cárdenas



Tipos de transparencia

Transparencia	Descripción
Acceso	Esconder diferencias en la representación de datos y como el recurso es accedido
Ubicación	Esconder el lugar donde se encuentra un recurso
Migración	Esconder que un recurso pudo moverse a otra ubicación
Reubicación	Esconder que un recurso puede moverse a otra ubicación mientras se encuentre en uso
Replicación	Esconder que un recurso es replicado.
Concurrencia	Esconder que un recurso puede ser compartido por varios usuarios
Falla	Esconder y recuperarse de la falla de un recurso

Lámina 13
Dr. Roberto Gómez Cárdenas



Apertura

Sistema que ofrece servicios de acuerdo a reglas estandarizadas que describen la sintaxis y semánticas de dichos servicios.





Lámina 14
Dr. Roberto Gómez Cárdenas



Objetivo apertura

- Que sea posible intercambiar componentes de software y hardware, dando a un usuario mayor posibilidad de escoger productos de acuerdo a sus necesidades y fomentando la competencia entre proveedores, que deben mejorar sus servicios para ganar clientes (ciudadanos).
- Es la plataforma adecuada para el desarrollo de aplicaciones distribuidas, porque se pueden combinar las ventajas de diferentes máquinas y sistemas operacionales.

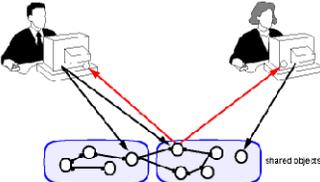


Lámina 15

Dr. Roberto Gómez Cárdenas



Los IDL

- En sistemas distribuidos los servicios generalmente son especificados a través de interfaces, que son descritas a través de una Interface Definition Language.
- Las IDLs solo capturan la sintaxis de los servicios.
- Especifican los nombres y funciones, tipos de parámetros, valores de regreso y excepciones que se pueden alcanzar, que se encuentran disponibles.

Lámina 16

Dr. Roberto Gómez Cárdenas



Ejemplo IDL en CORBA

```

module StockObjects {
    struct Quote {
        string symbol;
        long at_time;
        double price;
        long volume;
    };

    exception Unknown{};

    interface Stock {
        // Returns the current stock quote.
        Quote get_quote() [raises(Unknown)];

        // Sets the current stock quote.
        void set_quote(in Quote stock_quote);

        // Provides the stock description,
        // e.g. company name.
        readonly attribute string description;
    };
}
    
```

Estructura de datos

Interfaz

Excepción

Lámina 17

Dr. Roberto Gómez Cárdenas



Escalabilidad



Capacidad del sistema para mantener, si no mejorar, su rendimiento medio conforme aumenta o disminuye el número de clientes o recursos.



Lámina 18

Dr. Roberto Gómez Cárdenas

 **Escalamiento horizontal**

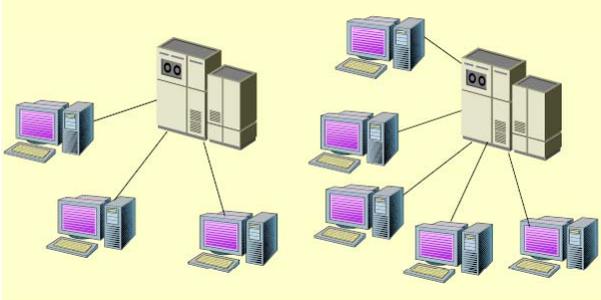


Lámina 19 Dr. Roberto Gómez Cárdenas

 **Escalamiento vertical**

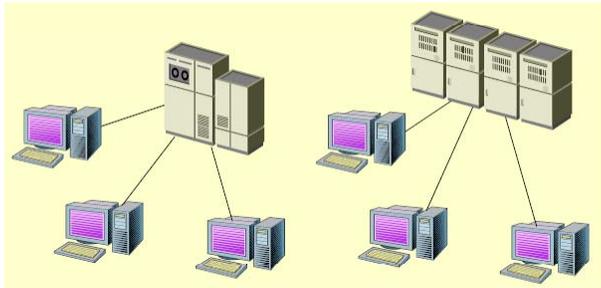


Lámina 20 Dr. Roberto Gómez Cárdenas



Técnicas de escalamiento

- Esconder las latencias de la comunicación
- Distribución
- Replicación

Lámina 21

Dr. Roberto Gómez Cárdenas

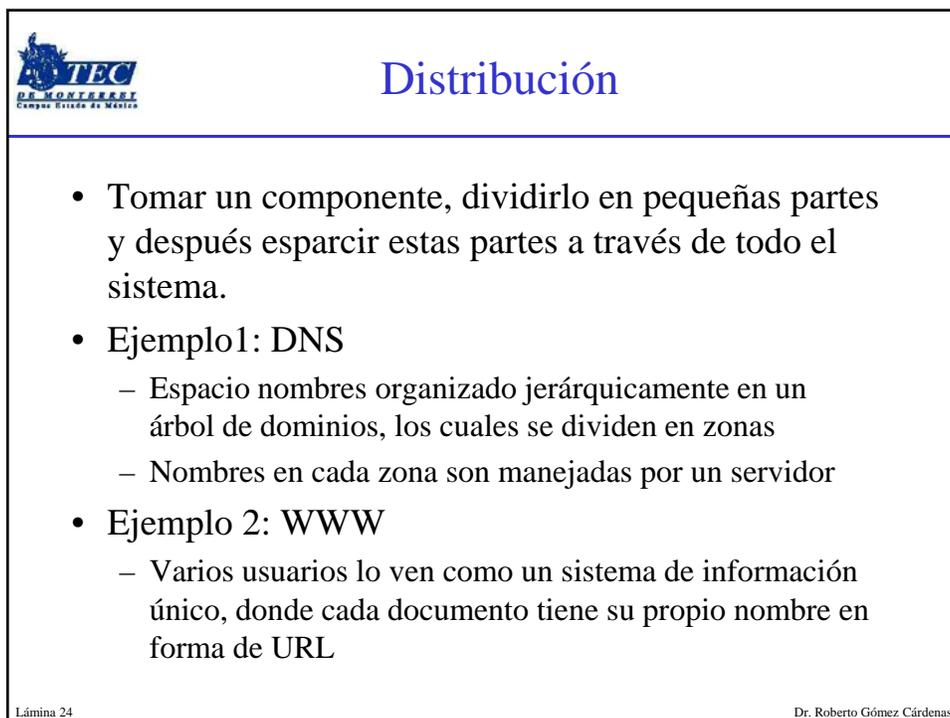
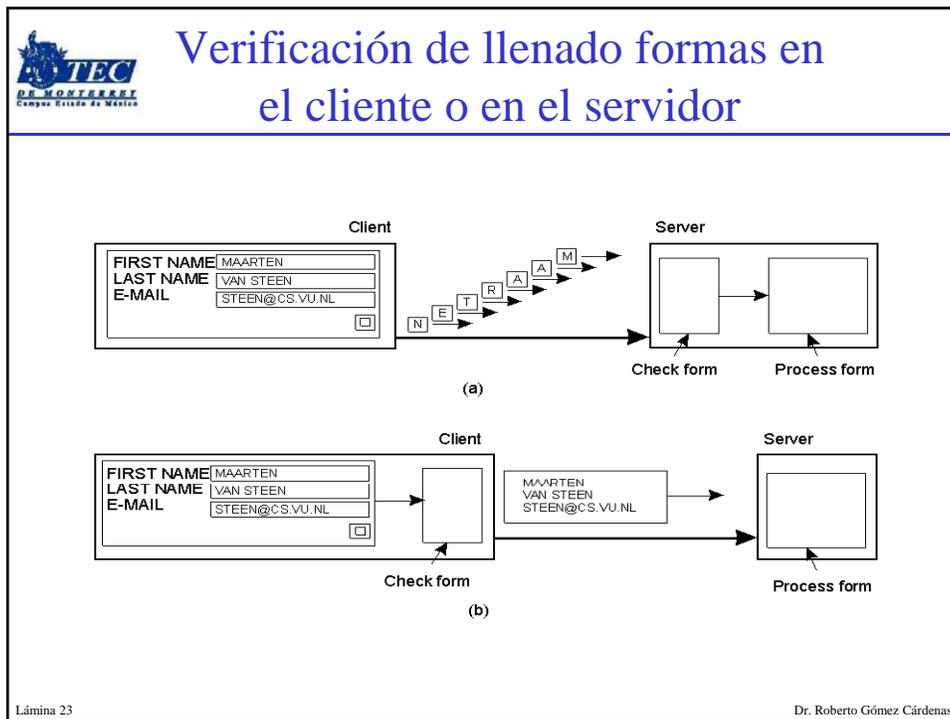


Escondiendo las latencias

- Intentar evitar la espera de respuestas a peticiones remotas de servicios.
- Una alternativa: uso de comunicaciones asíncronas.
- Otra alternativa: reducción de la comunicación entre las dos partes, pasándole trabajo al cliente.
 - Ejemplo: acceder bases de datos usando formas.

Lámina 22

Dr. Roberto Gómez Cárdenas





Ejemplo DNS

- Encontrando el nombre: *nl.vu.cs.flits*
 - Primero se busca en Z1, que regresa servidor de Z2...

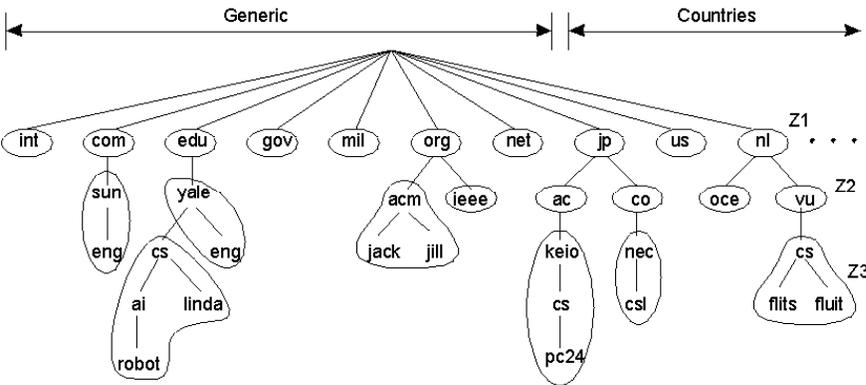


Lámina 25
Dr. Roberto Gómez Cárdenas



Replicar

- Replicar componentes a través del sistema distribuido.
- Incrementa disponibilidad y ayuda en el balanceo de cargas entre componentes para un mejor desempeño.
- Dos puntos a tomar en cuenta
 - Cache: forma especial de replicación
 - Consistencia: todas las replicas deben contar con el mismo contenido.

Lámina 26
Dr. Roberto Gómez Cárdenas



TEC
DE MONTERREY
Campus Estado de México

Distribución vs Replicación

- Distribución
 - Para obtener la totalidad de la información se tiene que consultar a todo el mundo

$$I = (x_1 + x_2 + \dots x_i \dots + x_n)$$

- Replicación
 - Asegurar la coherencia mutua de las copias a todo instante las copias físicas xi tengan el mismo valor x

$$x_1 = x_2 = \dots x_i \dots = x_n = x$$

Lámina 27

Dr. Roberto Gómez Cárdenas



TEC
DE MONTERREY
Campus Estado de México

Ejemplo distribución



X_1

X_2

X_3

Lámina 28

Dr. Roberto Gómez Cárdenas

 **Ejemplo replicación**



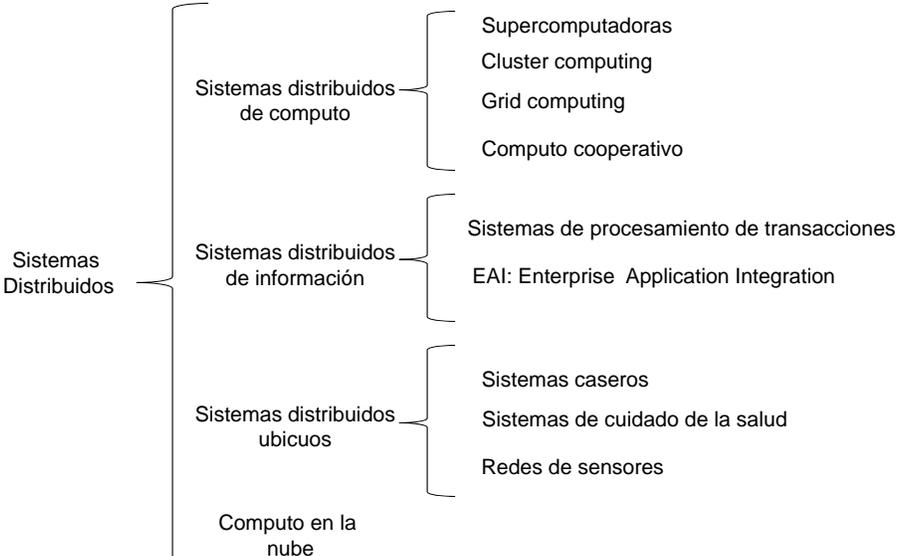
X1 Pierre 500
Sam 700
Juan 900

X2 Pierre 500
Sam 700
Juan 900

X3 Pierre 500
Sam 700
Juan 900

Lámina 29 Dr. Roberto Gómez Cárdenas

 **Tipos de sistemas distribuidos**



- Sistemas Distribuidos
 - Sistemas distribuidos de computo
 - Supercomputadoras
 - Cluster computing
 - Grid computing
 - Computo cooperativo
 - Sistemas distribuidos de información
 - Sistemas de procesamiento de transacciones
 - EAI: Enterprise Application Integration
 - Sistemas distribuidos ubicuos
 - Sistemas caseros
 - Sistemas de cuidado de la salud
 - Redes de sensores
 - Computo en la nube

Lámina 30 Dr. Roberto Gómez Cárdenas



Sistemas distribuidos de computo

- Usados para tareas de computo de alto desempeño.
- En esta categoría se pueden mencionar:
 - Supercomputadoras
 - Cluster computing
 - Grid computing
 - Computo cooperativo

Lámina 31

Dr. Roberto Gómez Cárdenas



¿De que se trata?

- El objetivo es solucionar un problema proporcionando pequeñas partes del problema a varias unidades de cálculo para solucionarlos y después combinar las soluciones parciales para encontrar una solución del problema.

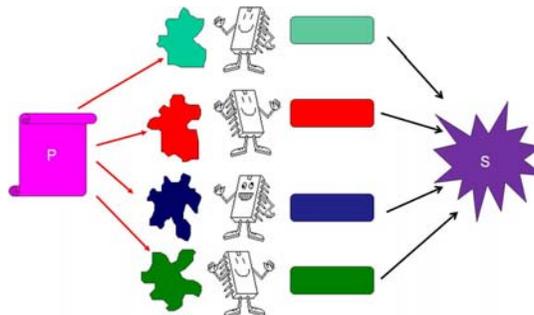


Lámina 32

Dr. Roberto Gómez Cárdenas



Computo distribuido vs supercomputadora

- Supercomputadora
 - Conjunto de procesadores fuertemente acoplados.
 - Soporte de algoritmos inter-procesadores.
 - Generalmente con un p específico.
 - Alto costo
- Computo distribuidos
 - Acoplamiento débil
 - Costo variable
 - Diferentes opciones en software




Lámina 33
Dr. Roberto Gómez Cárdenas



Lista de supercomputadoras más rápidas

Rank	Rmax Rpeak (Tflops)	Name	Computer Processor cores	Maker	Site Country, Year
1	1105 1456.7	<i>Roadrunner</i>	BladeCenter QS22LS21 129600 (Cell/Opteron)	IBM	Los Alamos National Laboratory United States, 2008
2	1059 1381.4	<i>Jaguar</i>	Cray XT5 150152 (Opteron)	Cray	Oak Ridge National Laboratory United States, 2008
3	487.01 608.83	<i>Pleiades</i>	SGI Altix ICE 8200EX 51200 (Xeon), InfiniBand	SGI	NASA/Ames Research Center United States, 2008
4	478.2 596.4	<i>Blue Gene/L</i>	eServer Blue Gene Solution 212992 (Power)	IBM	Lawrence Livermore National Laboratory United States, 2007
5	450.3 557.1	<i>Intrepid⁴³¹</i>	Blue Gene-P Solution 163840 (Power)	IBM	Argonne National Laboratory United States, 2007
6	433.2 579.38	<i>Ranger</i>	Sun Constellation System 82976 (Opteron), Infiniband	Sun	Texas Advanced Computing Center United States, 2008
7	266.3 355.51	<i>Franklin</i>	Cray XT4 38642 (Opteron)	Cray	NERSC/LBNL United States, 2008
8	205.0 260.2	<i>Jaguar</i>	Cray XT4 30976 (Opteron)	Cray	Oak Ridge National Laboratory United States, 2008
9	204.2 284.0	<i>Red Storm</i>	Cray XT3 38208 (Opteron)	Cray	NNSA/Sandia National Laboratories United States, 2008
10	180.6 233.47	<i>Dawning 5000A</i>	Dawning 30720 (Opteron)	Dawning	Shanghai Supercomputer Center People's Republic of China, 2008





Lámina 34
Dr. Roberto Gómez Cárdenas



Los clusters

- Conjunto de recursos de hardware reunidos para llevar a cabo una tarea en común.
- Características
 - Mejor rendimiento.
 - Disponibilidad.
 - Menor costo a una supercomputadora.
 - Alta disponibilidad





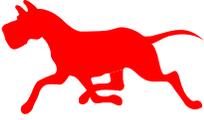
Lámina 35

Dr. Roberto Gómez Cárdenas



Diferencia clusters vs supercomputadoras

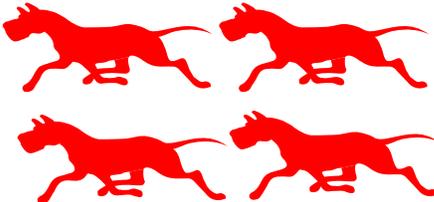
- Escalamiento
- Disponibilidad
- Administración sistema



Un perro



Un perro multícabezas



Un conjunto de perros

Lámina 36

Dr. Roberto Gómez Cárdenas



Grid computing

- *Grid* es una infraestructura de software y hardware para coordinar el proceso de compartir recursos y resolver problemas en organizaciones virtuales dinámicas.



The diagram illustrates the Grid Computing architecture. At the center is the 'Organización Virtual' (Virtual Organization), which includes 'Administración de cargas de trabajo' (Workload Management), 'Administración de Org. Virtual' (Virtual Organization Management), 'Servicios de Información' (Information Services), and 'Servicios de Supervisión' (Monitoring Services). This central organization is connected to 'Proveedores de recursos' (Resource Providers) on the left and 'Consumidores de recursos' (Resource Consumers) on the right. The consumers interact through a 'Portal Web de Grid' (Grid Web Portal) and 'Cliente de Acceso' (Access Client). A 'Portal de Acceso' (Access Portal) is also shown. The entire system is supported by a network infrastructure including a satellite and various server racks.

Lámina 37 Dr. Roberto Gómez Cárdenas



Computo voluntario

- Cómputo en el cual la gente (voluntarios) otorgan recursos computacionales a uno o varios proyectos.
 - Computo Distribuido.
 - Medio de almacenamiento.
- Características:
 - Los voluntarios son miembros del público general que tienen su computadora conectada a Internet.
 - Los proyectos generalmente son de tipo académico y de investigación científica (SETI@home, LHC@home, etc).

Lámina 38 Dr. Roberto Gómez Cárdenas



¿Y cuál es el problema?

- ¿Cómo los comunico?
- ¿Cómo los sincronizo?
- No todos los algoritmos se pueden paralelizar.



Lámina 39



Dr. Roberto Gómez Cárdenas



Paralelización de instrucciones básicas

- Ejemplo: la formula del “chicharronero”

$$X = (-b + (b^2 - 4 * a * c)^{0.5}) / (2 * a)$$

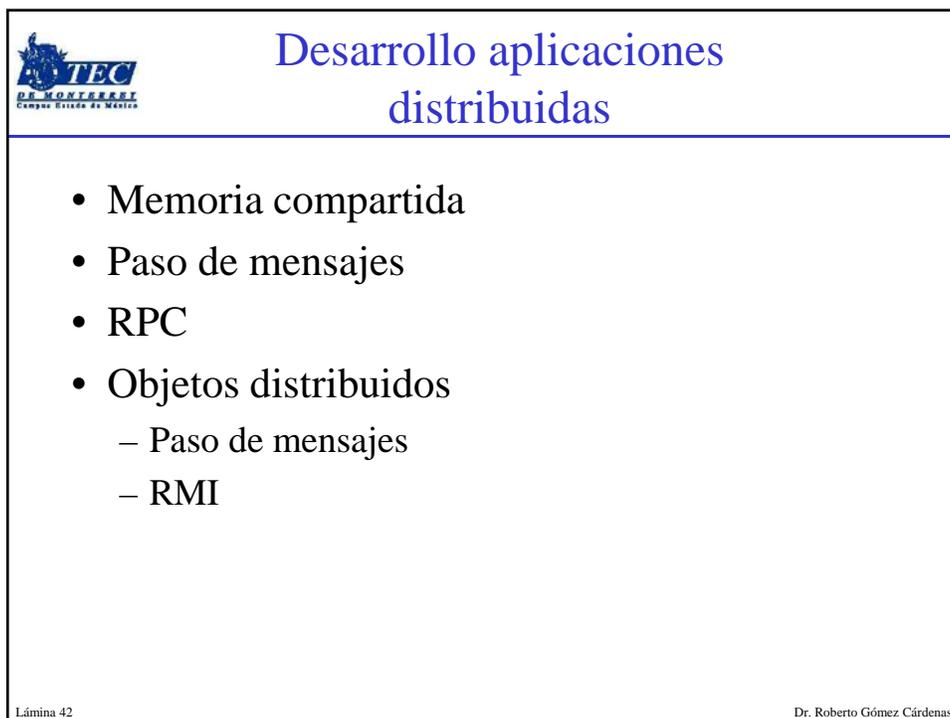
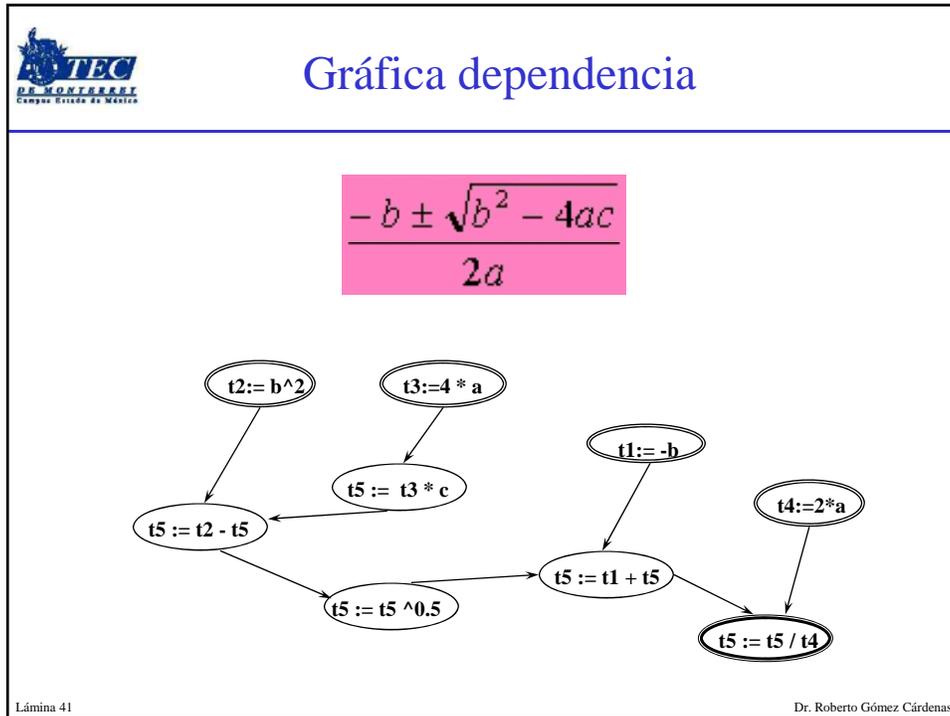
(1) $t1 := b^2$
 (2) $t2 := 4 * a$
 (3) $t2 := t2 * c$
 (4) $t1 := t1 - t2$
 (5) $t1 := t1^{0.5}$
 (6) $t1 := -b + t1$
 (7) $t2 := 2 * a$
 (8) $t1 := t1 / t2$

(1) $t1 := -b$
 (2) $t2 := b^2$
 (3) $t3 := 4 * a$
 (4) $t4 := 2 * a$
 (5) $t5 := t3 * c$
 (6) $t5 := t2 - t5$
 (7) $t5 := t5^{0.5}$
 (8) $t5 := t1 + t5$
 (9) $t5 := t5 / t4$

$$\frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Lámina 40

Dr. Roberto Gómez Cárdenas





Memoria compartida

- No es el caso de sistemas distribuidos, aplica a lo que es computadoras paralelas.
- Procesadores comparten una zona de memoria.
- Comunicación y sincronización se llevan a cabo a través de esta zona.

Lámina 43

Dr. Roberto Gómez Cárdenas



Paso mensajes

- Procesadores se envían mensajes entre ellos.
- Una opción: sockets

```
main()
{
    sd = socket(AF_UNIX, SOCK_STREAM, 0);
    strcpy(sin.sun_path, "./socket");
    sin.sun_family = AF_UNIX;
    bind(sd, (struct sockaddr *)&sin, sizeof(sin));
    listen(sd, 5);
    while (1) {
        sd_actual = accept(sd, 0, 0);
        if ( fork() == 0 ) {
            atencion(sd_actual);
            exit(0);
        }
    }
    close(sd);
    unlink("./socket");
}
```

Lámina 44

Dr. Roberto Gómez Cárdenas



RPC

- Enmascarar el envío mensajes
- Cliente llama función definida en servidor

```
int main (int argc, char *argv[])
{
    CLIENT *clnt;
    char *host;
    int a;
    int *per;
    double *sup;

    a = atoi(argv[1]);
    host = argv[2];
    clnt = clnt_create (host, CUADRO_PROG, CUADRO_VERS, "udp");
    if (clnt == NULL) {
        clnt_pcreateerror (host);
        exit (1);
    }
    per = perimetro_1(&a, clnt);
    sup = area_1(&a, clnt);
    printf("El perimetro del cuadrado es %d \n", *per);
    printf("El area del cuadrado es %d \n", *sup);

    clnt_destroy(clnt);
}
```

Lámina 45 Dr. Roberto Gómez Cárdenas



Vayamos más arriba

- PVM
 - Parallel Virtual Machine
- MPI
 - Message Passing Interface

Lámina 46 Dr. Roberto Gómez Cárdenas



PVM



- Parallel Virtual Machine
- Herramienta que permite crear aplicaciones paralelas y distribuidas.
- Funciona sobre una colección heterogénea de computadoras corriendo Unix en una o más redes.
- Comunicación: paradigma de paso de mensajes.
- Se basa en el concepto de Máquina Paralela Virtual
- La portabilidad prima sobre la potencia
- Tolerancia a fallas
- 1989 Oak Ridge National Laboratory
- <http://www.csm.ornl.gov/pvm/>

Lámina 47
Dr. Roberto Gómez Cárdenas



MPI



- Message Passing Interface
- Data de 1993
- Diseñado para un desempeño alto tanto en máquinas paralelas como en clusters de estaciones de trabajo.
- La especificación detalla las funciones que se pueden utilizar, no el modo como se compilan y lanzan-ejecutan los programas, lo cual puede variar de una implementación a otra.
- Siguiendo el modelo SPMD, el usuario escribirá su aplicación como un proceso secuencial del que se lanzarán varias instancias que cooperan entre sí.
- <http://www.mcs.anl.gov/research/projects/mpi/>

Lámina 48
Dr. Roberto Gómez Cárdenas



Sistemas distribuidos de información

- Organizaciones confrontadas con una gran variedad de aplicaciones de red, pero para las cuales resulta muy difícil inter-operar.
- Dos enfoques
 - Servidores corriendo aplicaciones accesibles a programas remotos conocidos como clientes.
 - Varios clientes realizan diferentes peticiones de forma concurrente
 - Transacciones distribuidas
 - Permitir que las aplicaciones se comuniquen entre ellas de forma directa: EAI

Lámina 49 Dr. Roberto Gómez Cárdenas



Sistemas procesamiento transacciones

- Sistemas usan transacciones requieren primitivas especiales que deben ser proporcionadas por el sistema distribuido o por el sistema del lenguaje.
- Ejemplos primitivas

Primitiva	Descripción
BEGIN_TRANSACTION	Marcar el inicio de una transacción
END_TRANSACTION	Terminar transacción e intentar completar esta
ABORT_TRANSACTION	Matar la transacción y restablecer los antiguos valores
READ	Leer datos de un archivo, o tabla
WRITE	Escribir datos a un archivo o tabla

Lámina 50 Dr. Roberto Gómez Cárdenas



Propiedades transacciones: ACID

- Atomicidad
 - Desde el punto de vista externo la transacción es indivisible.
- Consistencia
 - Una transacción T transforma un estado consistente de la base de datos en otro estado consistente, aunque T no tiene por qué preservar la consistencia en todos los puntos intermedios de su ejecución.
- Aisladas (Isolated)
 - Transacciones concurrentes no deben interferir entre ellas.
 - Asegura que la realización de dos transacciones sobre la misma información sean independientes y no generen ningún tipo de error.
- Durables
 - Una vez que la transacción es completada (commits), los cambios son permanentes.
 - Ninguna falla después de completada puede deshacer los resultados.

Lámina 51

Dr. Roberto Gómez Cárdenas



Enterprise Application Integration

- A medida que las aplicaciones son separadas de las bases de datos, más evidente se hace la necesidad de facilidades para integrar aplicaciones independiente de sus bases de datos.
- Los componentes de las aplicaciones deben poder comunicarse entre ellos y no necesariamente a través del modelo de petición/respuesta.

Lámina 52

Dr. Roberto Gómez Cárdenas



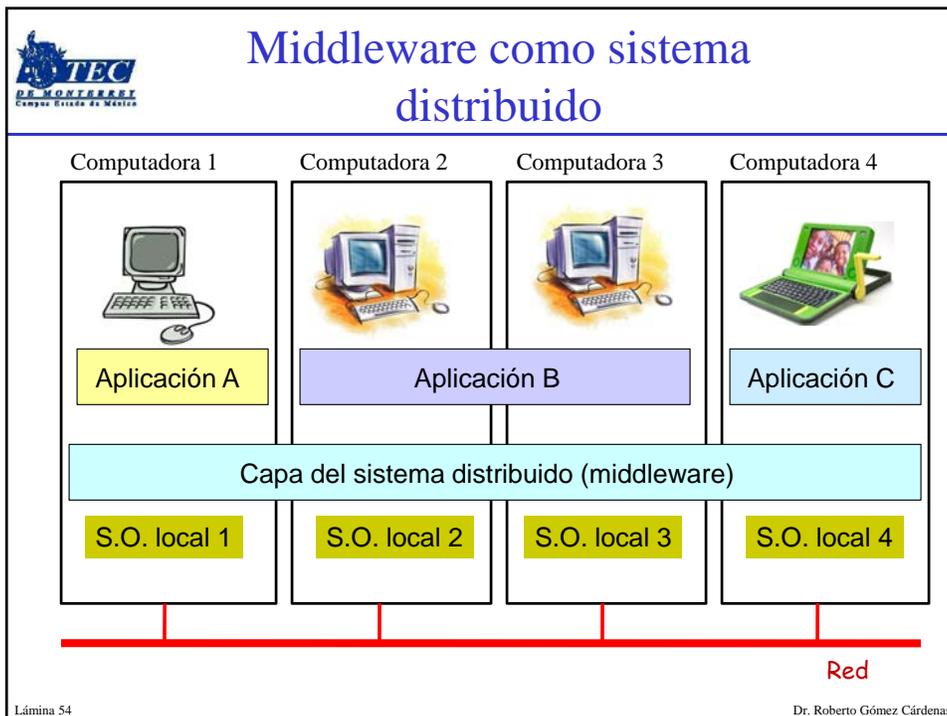
Middleware



Copyright Quino © 2001

Lámina 53

Dr. Roberto Gómez Cárdenas





Opciones middleware de comunicación

- Necesidad de comunicación entre aplicaciones lleva a diferentes modelos de middleware de comunicación.
 - RPC
 - Remote Procedure Call
 - RMI
 - Remote Method Invocation
 - Message Oriented Middleware (MOM)
 - Aplicaciones envían mensajes a los puntos de contacto
 - Mensajes descritos por un subject.
 - Sistemas de tipo publish/subscribe

Lámina 55

Dr. Roberto Gómez Cárdenas



El modelo Publish/Subscribe

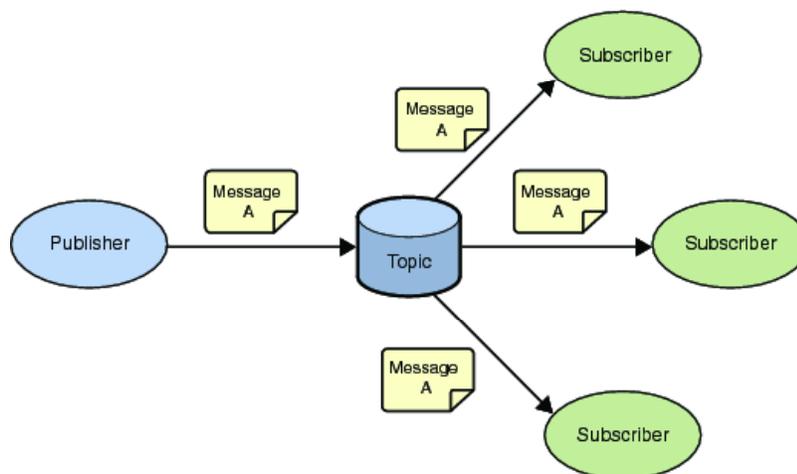


Lámina 56

Dr. Roberto Gómez Cárdenas

 **El modelo Point-to-Point**

- Modelo de mensajería tradicional

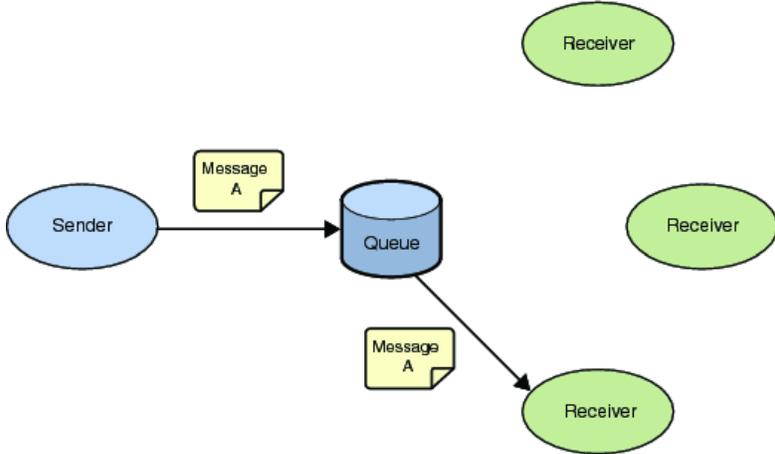


Lámina 57 Dr. Roberto Gómez Cárdenas

 **Sistemas distribuidos ubicuos**

- Dispositivos móviles y empotrados
- Pequeños, con batería, móviles y que cuenten con una conexión inalámbrica.
- Este tipo de sistemas son parte del ambiente que nos rodea.
- En algunos casos los dispositivos pueden ser configuradas por sus dueños, pero de otra forma ellos necesitan descubrir su ambiente e introducirse en él lo mejor posible.

Lámina 58 Dr. Roberto Gómez Cárdenas



Ejemplos redes ubicuoas

- Sistemas caseros.
- Sistemas de cuidado de la salud.
- Redes de sensores.

Lámina 59

Dr. Roberto Gómez Cárdenas



Sistemas caseros

- Una o más computadoras personales, TVs, equipos de audio y video, dispositivos de juego, teléfonos, PDAs.
- Algunos retos
 - Auto administración
 - Auto configuración
 - Espacios personales

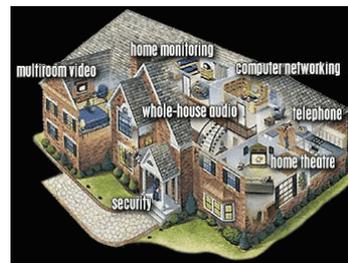
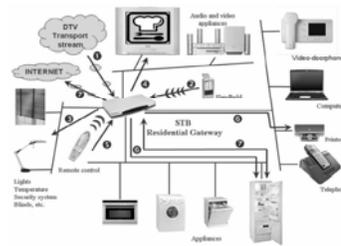


Lámina 60

Dr. Roberto Gómez Cárdenas

 **Home Entertainment Network**

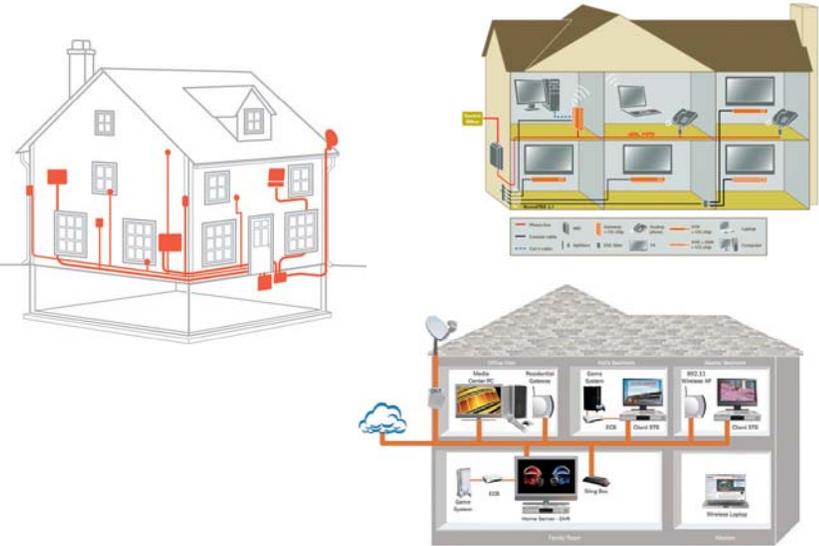


Lámina 61 Dr. Roberto Gómez Cárdenas

 **Sistema electrónicos de cuidado de salud**

- Sistemas relacionados con el cuidado de la salud de las personas.
- Dispositivos para monitorear la salud de las personas y solicitar ayuda de forma automática en caso necesario.
- Dispositivos con sensores organizados dentro de una BAN (Body Area Network)
- El sistema debe estorbar lo menos posible al usuario.
- El sistema debe funcionar mientras la persona se mueve sin cables atados a dispositivos no móviles.

Lámina 62 Dr. Roberto Gómez Cárdenas



Arquitecturas

1. Un hub central es parte de la red BAN y colecta los datos conforme se necesitan.
 - De tiempo en tiempo los datos son enviados a un dispositivo de almacenamiento más grande

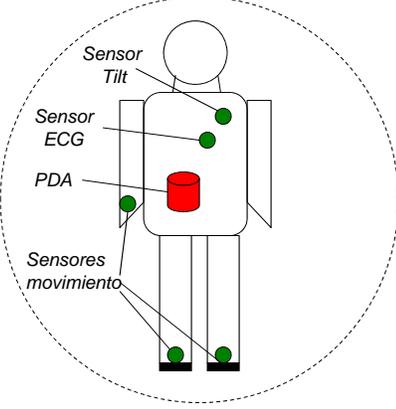
2. La BAN esta “atada” a una red externa, a través de una conexión wireless, que recibe los datos de los sensores.

Lámina 63
Dr. Roberto Gómez Cárdenas



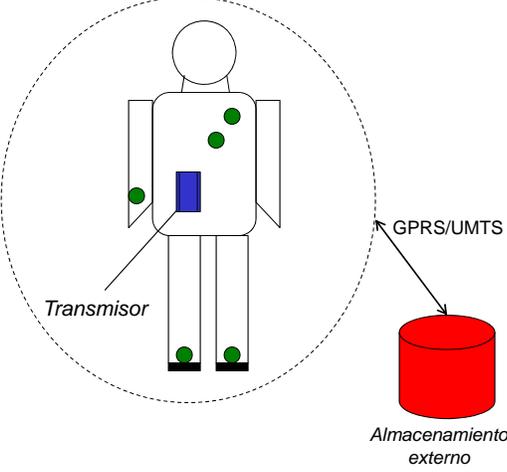
Esquema de las dos arquitecturas

Red del área del cuerpo



Tilt Sensor: Sensor de inclinación
ECG Sensor: Sensor electrocardiograma

Red del área del cuerpo



Almacenamiento externo

Lámina 64
Dr. Roberto Gómez Cárdenas



Redes de sensores

- Consiste de decenas a centenas o miles de nodos pequeños con un dispositivo sensorial.
- La mayor parte de los dispositivos usan comunicación inalámbrica y los nodos usan baterías.
- Limitación en recursos, capacidad de comunicación y energía hacen que la eficiencia sea un punto clave a tomar en cuenta en su diseño.

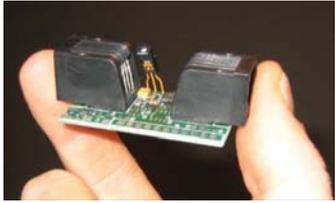
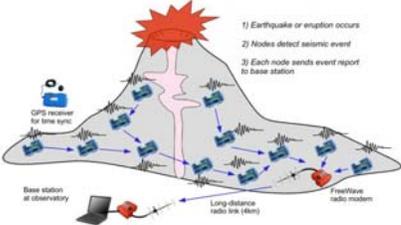
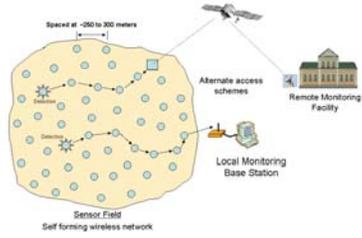



Lámina 65 Dr. Roberto Gómez Cárdenas



Redes sensoriales y sistemas distribuidos

- Redes sensoriales pueden ser vistas como una base datos distribuidas
 - Redes usadas para medir y supervisar aplicaciones.
 - Operador extrae información de la red, o parte de esta, a través de queries
- Operaciones de agregación
 - Una operación de agregación calcula un valor único a partir de una colección de valores.
 - Redes sensores: aggregation queries



Origen

2	7	5	1	6	8	3
---	---	---	---	---	---	---

Resultados

Sum	32
Max	8
...	

Lámina 66 Dr. Roberto Gómez Cárdenas



Sistemas SCADA

- Supervisory Control and Data Acquisition
- Consisten primordialmente de cinco componentes principales:
 - una computadora maestra,
 - una terminal remota,
 - periféricos para la comunicación,
 - software de aplicación y
 - transductores.

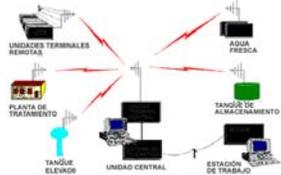




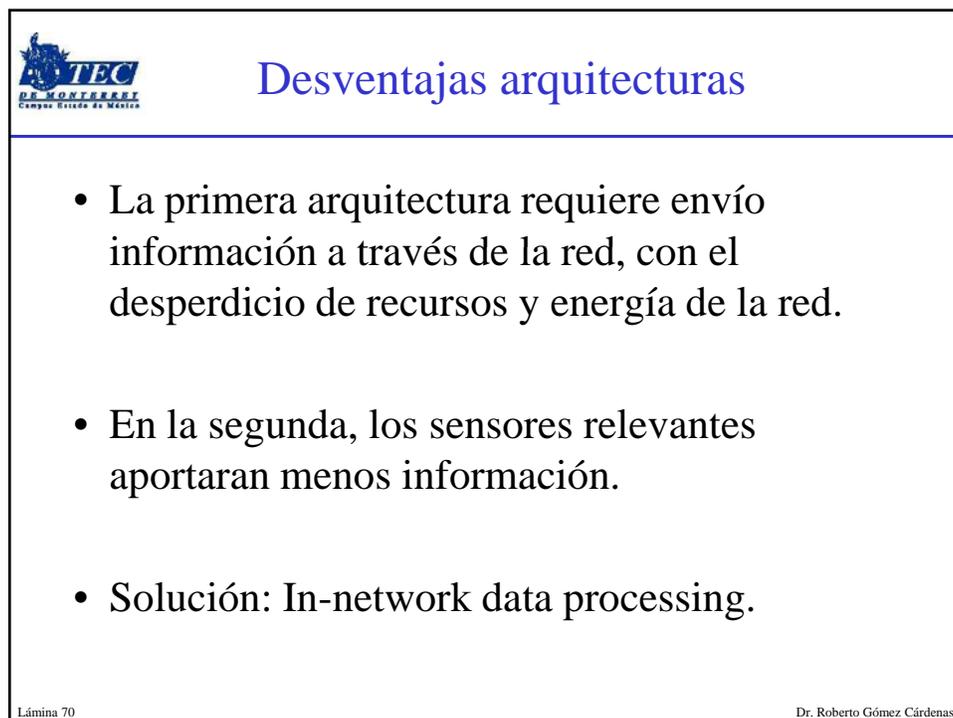
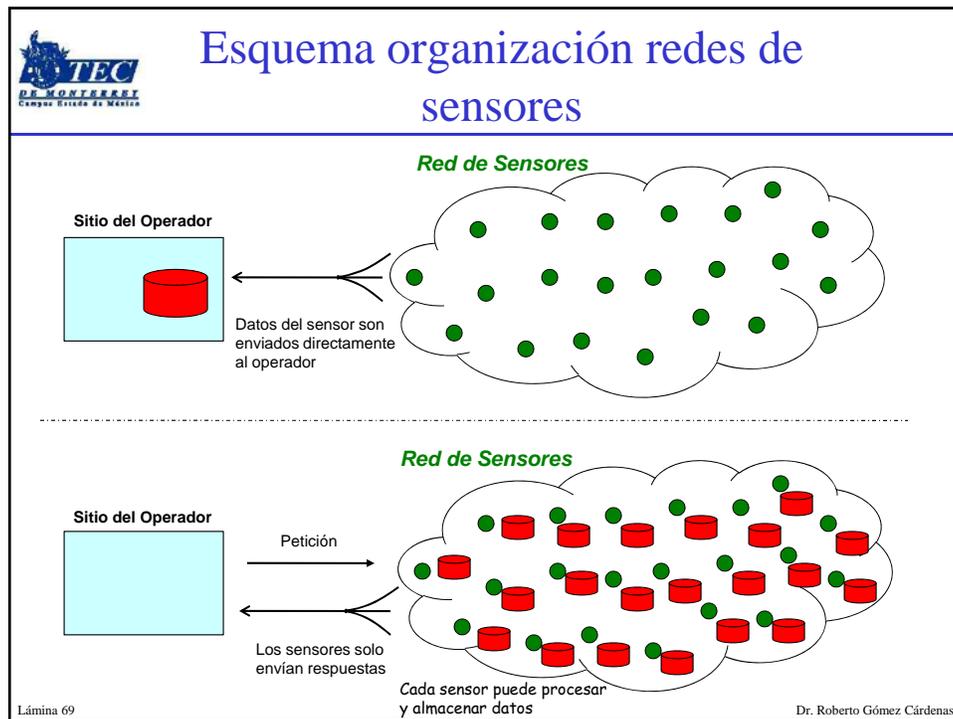
Lámina 67 Dr. Roberto Gómez Cárdenas



Organización redes sensoriales

- Para organizar una red de sensores como una base de datos distribuida, existen dos formas de hacerlo:
 - Sensores no llevan a cabo ningún cálculo, solo envían información a una base de datos centralizada ubicada en el sitio del operador.
 - Sensores envían información a sensores relevantes, estos calculan una respuesta y el operador *agrega* las respuestas recibidas.

Lámina 68 Dr. Roberto Gómez Cárdenas





In-network data processing

- Procesamiento puede hacerse de varias formas.
 - Enviar query a todos los procesadores a través de un árbol que abarque todos los nodos y cuya raíz sea el iniciador.
 - Agregaciones se hacen donde dos o más ramas de un árbol se unen.

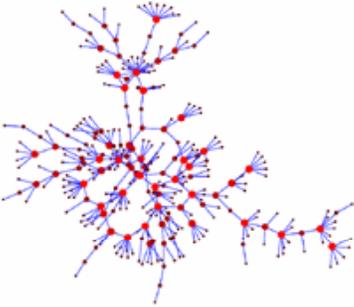


Lámina 71

Dr. Roberto Gómez Cárdenas



Arquitecturas vs In-network data processing

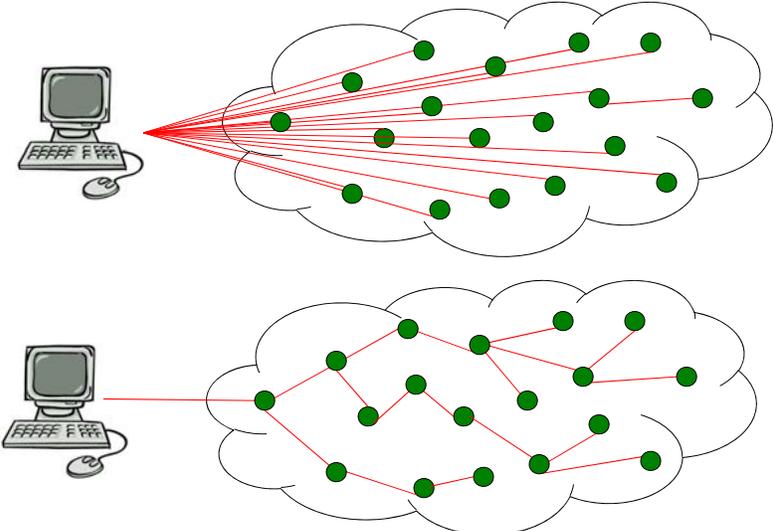


Lámina 72

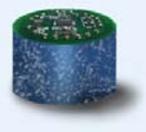
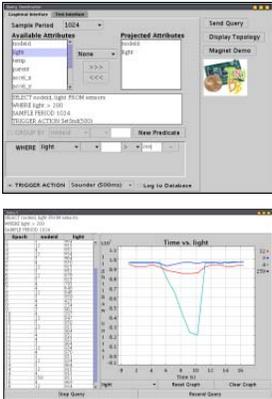
Dr. Roberto Gómez Cárdenas



TEC
DE MONTERREY
Campus Estado de México

Tiny DB

- Interface de base de datos declarativa para redes de sensores inalámbricas.
 - Interfaz tipo SQL
- Nodo intermediario colecta y agrega los resultados de sus hijos, junto con lo que ha recolectado y lo envía a la raíz.
- Para dar eficiencia, los queries abarcan un periodo de tiempo que permite la calendarización de operaciones, de tal forma que los recursos de red y energía sean aprovechados de forma óptima.
- ¿Queries iniciados en diferentes puntos?

Dr. Roberto Gómez Cárdenas

Lámina 73





Cloud computing

- Computación en nube
- Definición NIST (Draft del 6-1-09)
 - Es un modelo para habilitar acceso a un conjunto de recursos informáticos configurables (p.e. redes, servidores, almacenamiento, aplicaciones y servicios) que puede ser aprovisionados y liberados rápidamente con un mínimo esfuerzo o interacción con el proveedor de servicio.
- La nube debe promover *disponibilidad* y esta compuesta de cinco características esenciales, tres modelos de entrega y cuatro modelos de implantación.

Lámina 75 Dr. Roberto Gómez Cárdenas



Características esenciales

- On demand self-service
- Ubiquitous network access
- Location independent resource pooling
- Rapid elasticity
- Measured Service

Lámina 76 Dr. Roberto Gómez Cárdenas



On-demand self-service

- Consumidor puede unilateralmente aprovisionarse de recursos computacionales, como tiempo de servicio y almacenamiento en red, de forma automática conforme lo necesite sin requerir interacción humana con cada proveedor de servicio.
- El consumidor *pay-as-you-go* y solo usa lo que necesita, todo manejado por un browser, una aplicación o un API

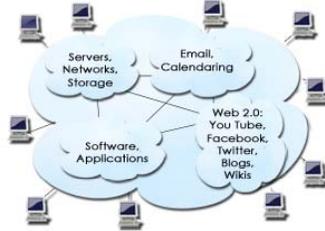


Lámina 77
Dr. Roberto Gómez Cárdenas



Acceso ubíquo a la red

- Los recursos se encuentran disponibles en la red y son accedidos a través de mecanismos que promueven el uso de plataformas heterogeneas simples.



Lámina 78
Dr. Roberto Gómez Cárdenas



Poleo de recursos independiente de su ubicación

- El proveedor de recursos usa un modelo de atención concurrente, con diferentes recursos físicos y virtuales dinámicamente asignados y reasignados de acuerdo a la demanda del consumidor.
- Consumidor no controla o no conoce la ubicación exacta de los recursos proporcionados, pero puede especificar una ubicación en un nivel de abstracción alto (país, estado o datacenter).



Lámina 79

Dr. Roberto Gómez Cárdenas



Elasticidad rápida

- Los recursos son proveerse rápida y elásticamente para escalar de forma inmediata tanto hacia arriba como hacia abajo.
- Para el consumidor, los recursos disponibles para su uso aparecen como infinitos y pueden ser adquiridos en cualquier cantidad y tiempo.



Lámina 80

Dr. Roberto Gómez Cárdenas



Servicio a la medida

- De forma automática se controlan y optimizan los recursos locales usados recursos ofrecidos de acuerdo al tipo de servicio (p.e. almacenamiento, procesamiento, ancho de banda y cuentas activas usuarios).
- El uso de los recursos puede ser monitoreado, controlado y reportado proporcionando transparencia del recurso consumido tanto para el proveedor como para el consumidor



Lámina 81

Dr. Roberto Gómez Cárdenas



Modelos de entrega

- Cloud Software as a Service (SaaS).
 - Uso de aplicaciones corriendo en la nube, accesibles desde varios clientes a través de interfaces “delgadas”.
- Cloud Platform as a Service (PaaS).
 - Desarrollar dentro de la nube aplicaciones usando lenguajes de programación y herramientas soportadas por el proveedor.
- Cloud Infrastructure as a Service (IaaS).
 - Proporciona procesamiento, almacenamiento, redes y otros recursos computacionales donde el consumidor puede desarrollar y correr cualquier software.

Lámina 82

Dr. Roberto Gómez Cárdenas



Modelos de implementación

- Nube privada
 - La nube es operado solo para una organización.
 - Puede ser manejada por la misma organización o un tercero.
- Nube comunitaria
 - La nube es compartida por varias organizaciones y soporta a una comunidad específica que tiene intereses comunes (p.e. misión, requerimientos de seguridad, políticas y consideraciones de compliance).
 - Puede ser manejada por la misma organización o un tercero.

Lámina 83

Dr. Roberto Gómez Cárdenas

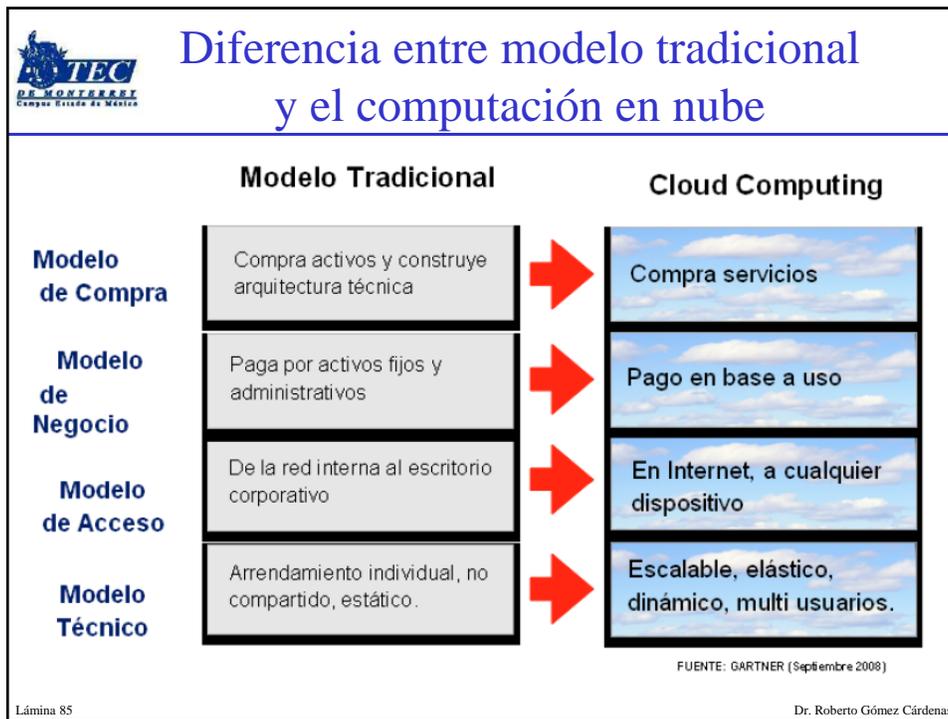


Modelos de implementación

- Nube pública
 - Disponible a todo el público y o a un grupo industrial grande y es propiedad de una organización que vende servicios de nube.
- Nube híbrida.
 - La infraestructura de la nube esta compuesta por dos o más nubes (privada, comunitaria o pública) que permanecen como entidades únicas pero que son unidos por tecnología estandarizada o propietaria que habilita la portabilidad de datos y aplicaciones (p.e. cloud bursting).

Lámina 84

Dr. Roberto Gómez Cárdenas



 **Amazon y Google**



Lámina 87 Dr. Roberto Gómez Cárdenas

 **Arquitecturas sistemas distribuidos**

Lámina 88 Dr. Roberto Gómez Cárdenas



Arquitecturas sistemas distribuidos

- Arquitectura de un sistema
 - ¿Qué componentes lo forman?
 - ¿Cómo interaccionan entre sí?

- Dos modelos principales en los sistemas distribuidos
 - Cliente-servidor
 - Igual a igual (peer-to-peer)

Lámina 89
Dr. Roberto Gómez Cárdenas



Arquitectura igual a igual

- Los procesos tienen un rol similar, aunque pueden asumir un rol cliente/servidor en ciertos momentos.
- Mejora la tolerancia a falla y la escalabilidad
- Difícil de coordinar
- Ejemplo:
 - Búsqueda distribuida,
 - Enrutamiento
 - Cálculo distribuido



Elegir una red inalámbrica

Haga clic en cualquier elemento de la siguiente lista para conectarse a una red inalámbrica en el alcance o para obtener más información.

 Free Public WiFi Red de equipo a equipo no segura	 Free Public WiFi Red de equipo a equipo no segura
---	---

Lámina 90
Dr. Roberto Gómez Cárdenas



Arquitectura cliente/servidor

- **Cliente**
 - Proceso que quiere acceder a datos, usar recursos o realizar operaciones en un dispositivo diferente.
- **Servidor**
 - Proceso que administra datos o cualquier otro tipo de recurso y que es capaz de hacerlos disponibles a otros procesos que están en otros dispositivos.
- **Interacción**
 - Enviar petición / recibir resultado

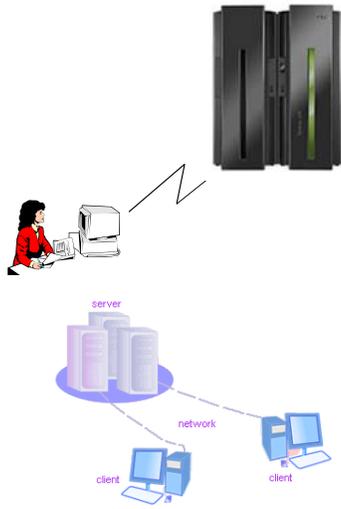


Lámina 91

Dr. Roberto Gómez Cárdenas



Interacción cliente servidor

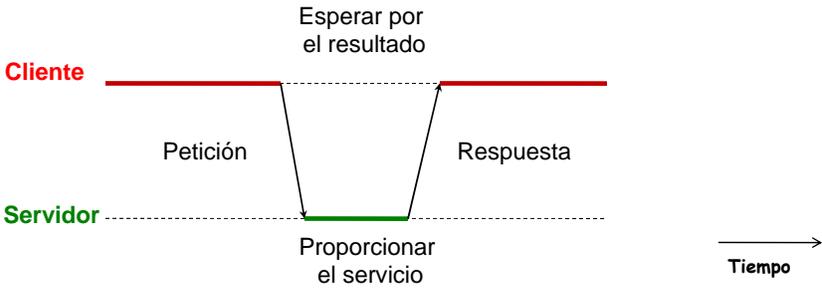
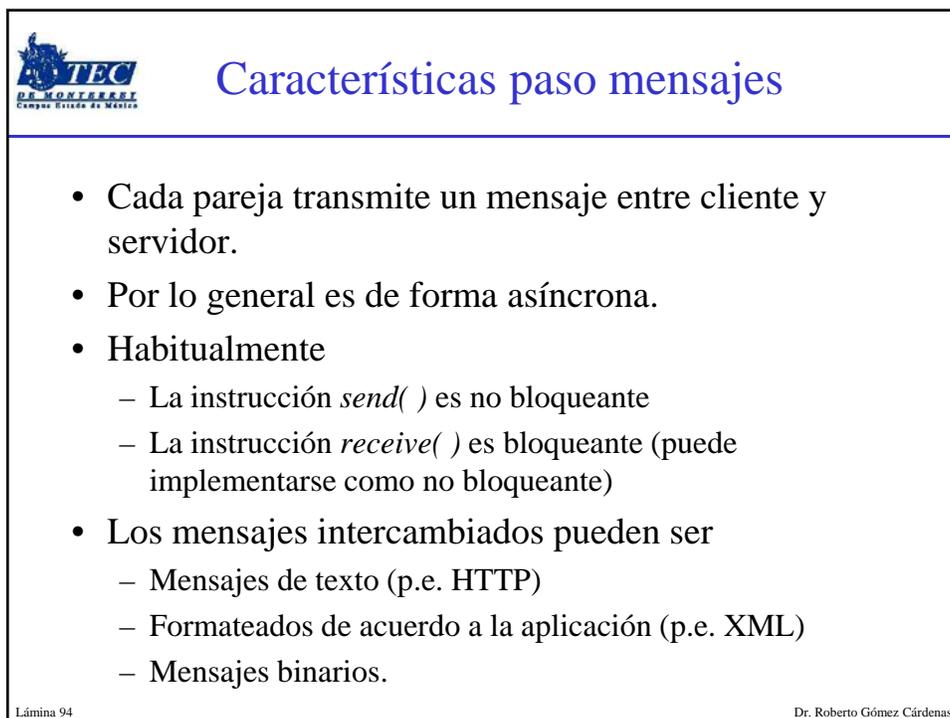
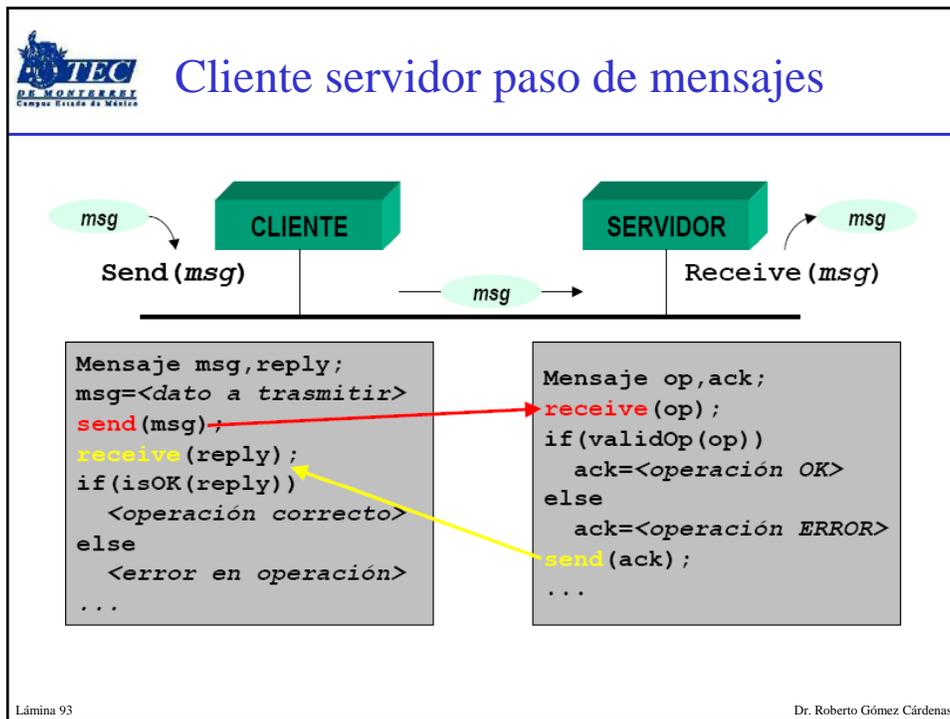
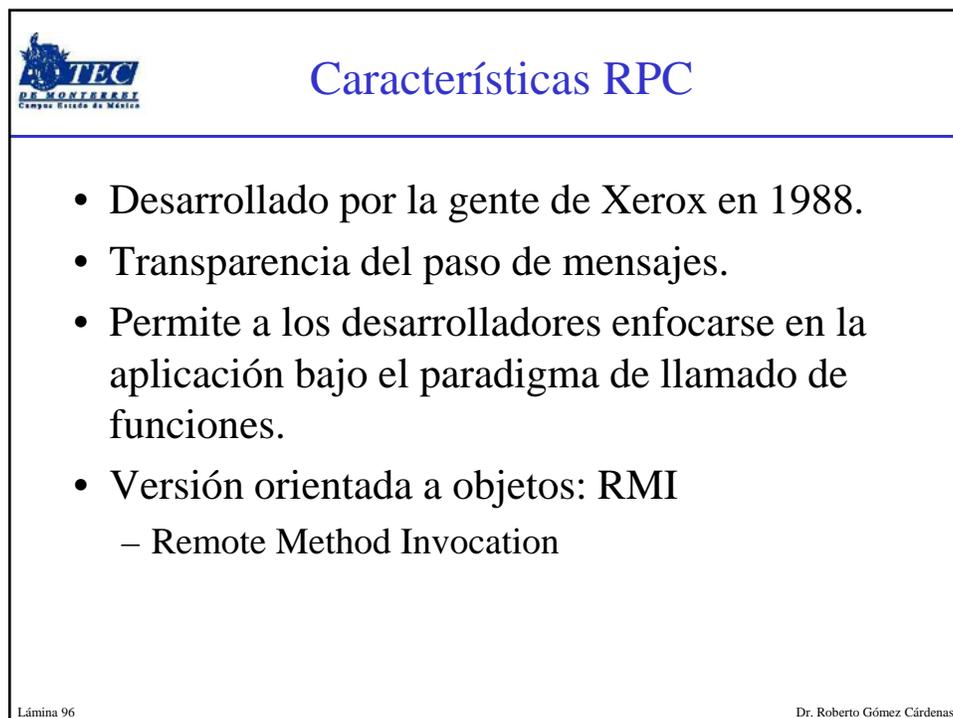
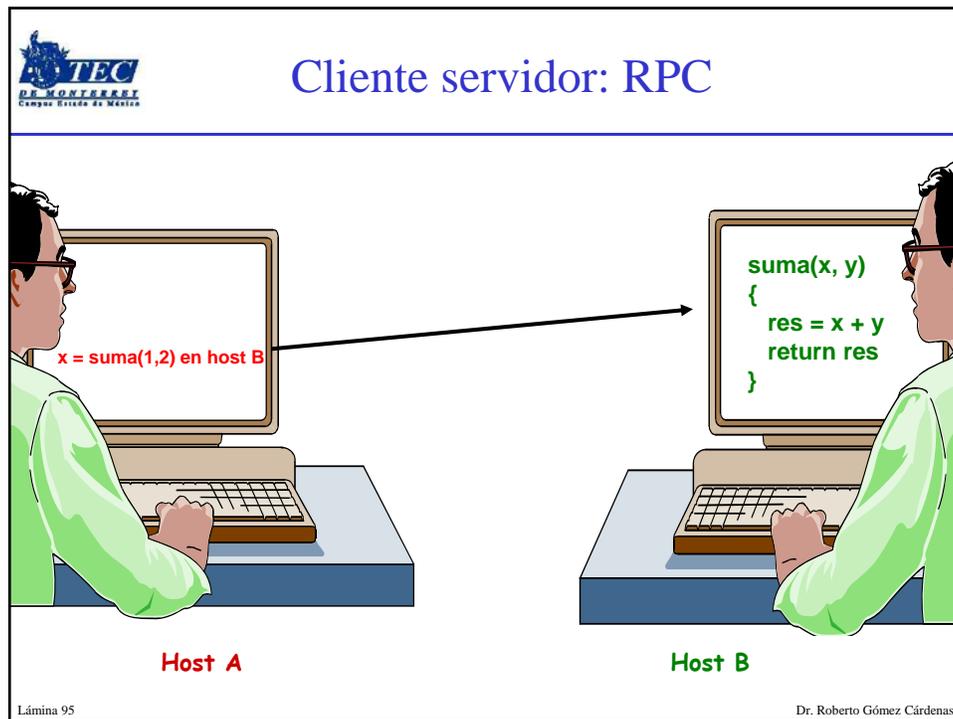


Lámina 92

Dr. Roberto Gómez Cárdenas







Tipos servidores

- Servidor de archivos
- Servidor de impresoras
- Servidor de red
- Servidor de bases de datos
- Servidor de aplicaciones
- Servidor de servicios de Internet
 - WWW, ftp, correo, etc

Lámina 97
Dr. Roberto Gómez Cárdenas



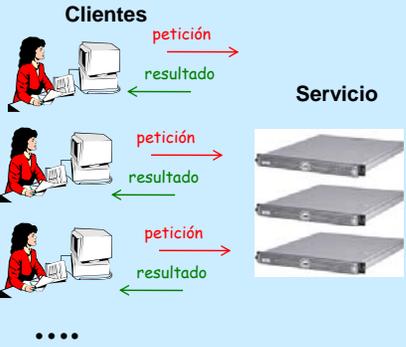
¿Un solo servidor o varios servidores?

Cientes



Un solo servidor

Cientes



Múltiples servidores

Lámina 98
Dr. Roberto Gómez Cárdenas



Características múltiples servidores

- Servicios son suministrados por múltiples servidores
 - Distribución de los recursos entre servidores
 - Ejemplo: Web
 - Mantenimiento de las ligas
 - Ligas perdidas
 - Replicación de los recursos
 - Servidores de alta disponibilidad: sitios de descarga, portales
- Replicación de recursos
 - Aumentar prestaciones
 - Tolerancia a fallas
 - Disponibilidad
 - Problema de coordinación o de coherencia
 - La información no llega simultáneamente a todos los servidores.

Lámina 99
Dr. Roberto Gómez Cárdenas



Rack servers vs Blade servers

- Una computadora moderna requiere de E/S Procesador y Memoria
- Rack
 - Los componentes son unidades completas que contiene la CPU, memoria, fuente de alimentación, ventiladores y disipadores.
 - Estos servidores son atornillados en el rack, y cada uno es conectado a la red corporativa usando un cable separado.
- Blade
 - Optimizados para minimizar el uso de espacio físico.
 - Incluye CPU, memoria y dispositivos para almacenar datos.
 - No tiene fuente de alimentación eléctrica ni ventiladores.
 - Son insertados en slots y enlazados entre si gracias a un bus de alta velocidad dentro del chasis.






Lámina 100
Dr. Roberto Gómez Cárdenas



Servidores pesados o clientes pesados

- Fat clients vs fat servers
- El modelo de servidor pesado carga más funcionalidades en el servidor
 - Groupware, traductores y servidores web son ejemplos de este tipo de servidores.
- El modelo de cliente pesado hace lo contrario.
 - Servidores de bases de datos son ejemplos de clientes pesados

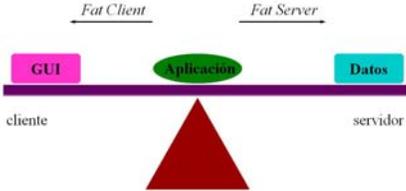


Lámina 101
Dr. Roberto Gómez Cárdenas



Características clientes pesados

- La forma más tradicional de la arquitectura cliente/servidor.
- La parte pesada de la aplicación corre en el cliente.
- En modelos de servidores de archivos y bases de datos, el cliente sabe como están los datos organizados y almacenados en el lado del servidor.
- Estos clientes son usados para soporte de decisiones y software personal.
- Proporcionan flexibilidad y oportunidades para crear herramientas de tipo front-end que los usuarios finales crean sus propias aplicaciones.

Lámina 102
Dr. Roberto Gómez Cárdenas



Características servidores pesados

- Más fáciles de administrar y liberar en la red.
 - Ya que la mayor parte del código corre en el servidor.
- Intentan minimizar intercambios de red creando niveles de servicio más abstractos.
- En lugar de exportar datos en bruto, exportan procedimientos (o métodos) que operan con los datos.
- El cliente proporciona el GUI e interactúa con servidor a través de RPCs (o RMI)

Lámina 103

Dr. Roberto Gómez Cárdenas



Server side scripting

- Es una tecnología de servidor web en la cual la petición de un usuario es atendida corriendo un script directamente en el servidor del web para generar páginas dinámicas
 - Sitios web interactivos que sirven de interfaz con bases de datos.
 - Escritos en lenguajes como Perl y PHP, son ejecutados por el servidor web cuando el usuario solicita un documento
 - El usuario no puede ver el código fuente de los scripts.

Lámina 104

Dr. Roberto Gómez Cárdenas



Client side scripting

- Generalmente se refiere a las clases de programas de computadoras en el web que son ejecutados en el lado del cliente, a través de su web browser
 - Es una parte importante del concepto de DHTML (Dynamic DHTML).
 - Se encuentran escritos en lenguajes como JavaScript o VBScript
 - El browser debe entender los scripts.
 - Es posible que los usuarios no puedan ver su código fuente

Lámina 105

Dr. Roberto Gómez Cárdenas



Arquitectura de una aplicación de n capas

- N capas (n-tiers)
- Cualquier número de capas
- Los más común:
 - 1 aplicación monolítica, 2 capas y 3 capas
- No implica separación física en diferentes dispositivos, pueden ser implementados
 - Físicamente (procesamiento distribuido)
 - Procesos en diferentes ubicaciones
 - Lógicamente
 - Procesos en un o más procesadores físicos.

Lámina 106

Dr. Roberto Gómez Cárdenas



Ventajas

- Permite modificar una capa sin tener que modificar toda la aplicación.
 - Aplicación una capa: existe un gran acoplamiento en la aplicación.
- Simplifica la administración de los sistemas.
- Disponibilidad inmediata de cambios en la lógica de aplicación o datos.
- Posibilita el balanceo de la carga de trabajo entre distintos dispositivos (computadoras).

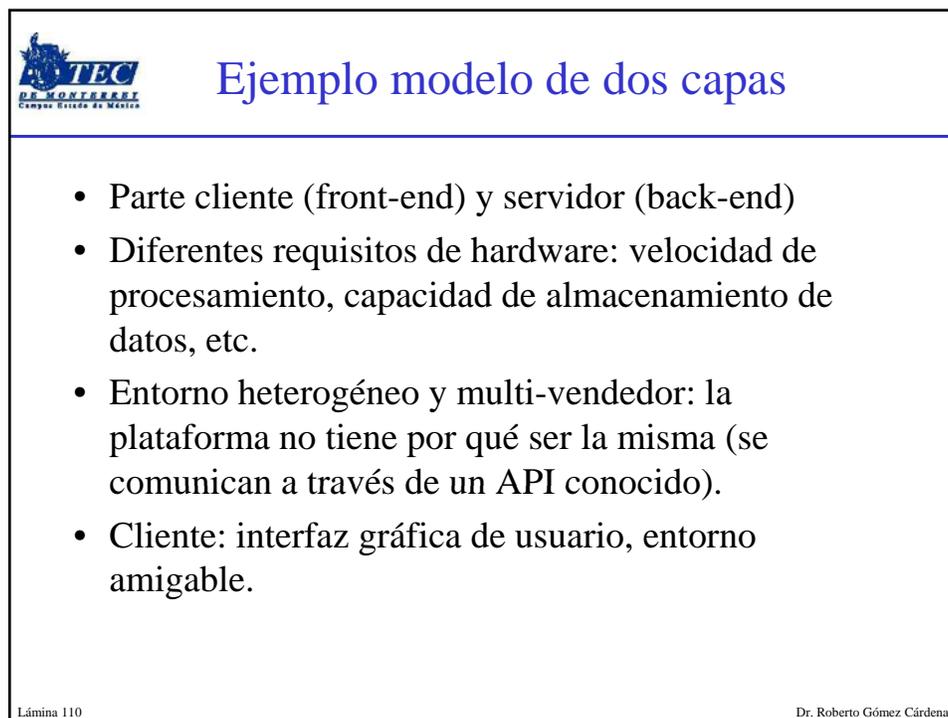
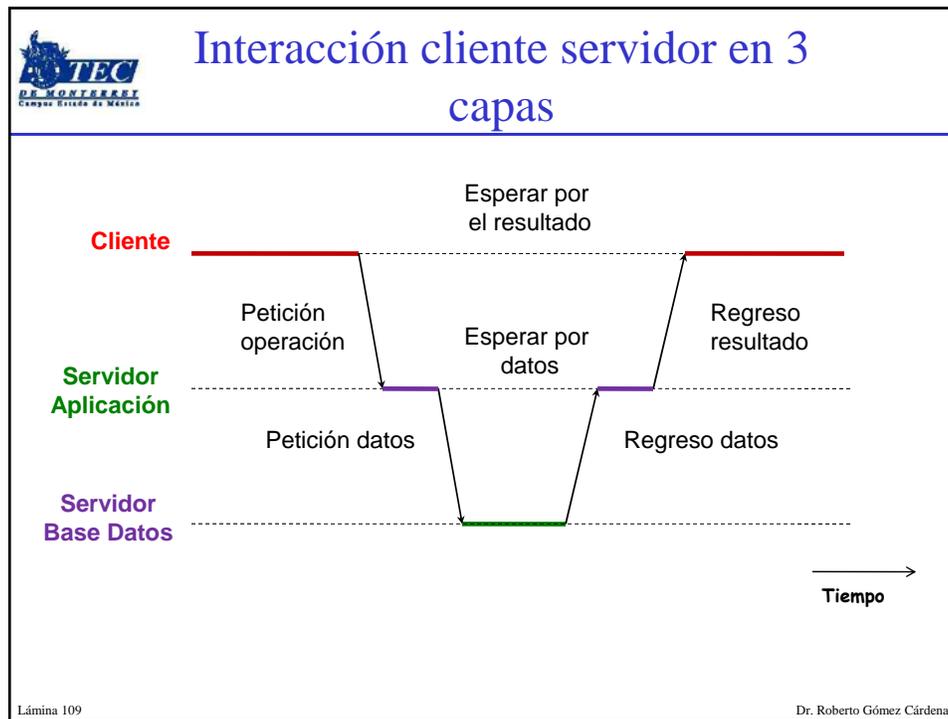
Lámina 107 Dr. Roberto Gómez Cárdenas



Ejemplo separación de funciones en modelo de 3 capas

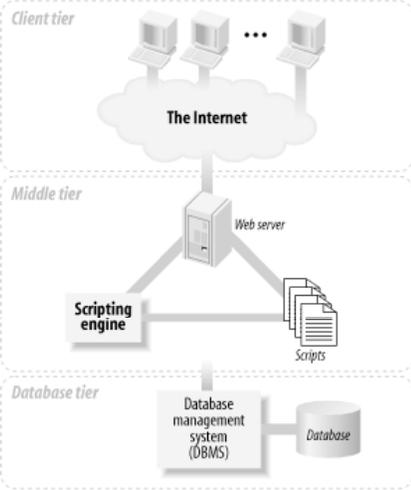
- Lógica de presentación
 - Interfaz de usuario a través de la cual interactúa con la aplicación.
 - Interfaz gráfico (GUI) o basado en texto.
- Lógica de negocio
 - El núcleo de la aplicación.
 - Implementa la funcionalidad de la aplicación.
- Lógica de datos
 - Administra la persistencia de los datos
 - Persistencia: Existencia o duración de una cosa durante mucho tiempo

Lámina 108 Dr. Roberto Gómez Cárdenas





Ejemplo modelo de tres capas



- Safari
- Firefox
- Internet Explorer
- Konqueror

- Apache
 - PHP (PHP Hypertext Pre-processor)
- Internet Information Services
 - ASP (Active Server Pages)

- MySQL
- SQL Server

Lámina 111

Dr. Roberto Gómez Cárdenas



Desventajas modulo 3 capas

- Asigna una gran carga a la red.
- Es más difícil programar y probar software que en una arquitectura de dos capas ya que más dispositivos tienen que comunicarse para completar una transacción del usuario.

Lámina 112

Dr. Roberto Gómez Cárdenas



Ejemplo aplicación



Amor, tus quesadillas ¿Con cuál salsa las quieres..?

Comercio electrónico

Lámina 113 Dr. Roberto Gómez Cárdenas