VENTAJAS Y DESVENTAJAS DE SISEMAS DISTRIBUIDOS EN 2001

Entre las principales ventajas con respecto a las computadoreas centralizadas, encontramos:

* Economía: Los microprocesadores ofrecen una mejor relación precio/rendimiento que las computadoras centrales
* Velocidad: Un sistema distribuido puede tener mayor poder de computo que una computadora centralizada individual
* Distribución inherente: Implica que un sistema distribuido puede emplear aplicaciones instaladas en computadoras remotas
* Confiabilidad: El sistema es consiente, aun si una computadora del sistema deja de funcionar
* Crecimiento proporcional: Cada vez que se requiere mayor poder de computo en el sistema, solo se pueden adicionar los incrementos de cómputo requerido. Debe ser flexible, modularizado

A pesar de los diferentes beneficios que introducen los sistemas distribuidos, todavía existen diferentes retos que deben ser resueltos como los siguientes.

* Software: Gran parte del software para sistemas distribuidos esta aun en desarrollo (Ya no es actual. Actualmente si hay sistemas distribuidos)

El mayor problema de software en nube, son los costos, saber administrarlos

* Redes: Los problemas de transmision en las redes de comunicación todavía son frecuentes en la trasferencia de grandes volúmenes de datos.

La conectividad es de los problemas mas grandes. Si no hay, no tienes nada

* Seguridad: Se necesitan mejores esquemas de protección para mejorar el acceso a información confidencial o secreta.

**SCL: Es para dar un cierto nivel de seguridad. Generando una conexión segura hasta cierto nivel**

* Tolerancia a fallas: Las fallas operativas y de componentes aun son frecuentes

En sistemas distribuidos se mide la latencia, que es el tiempo de respuesta del sistema.

Confiabilidad implica tolerancia al fallo, disponibilidad

Computo cuántico es lo que sigue de cómputo en la nube

**Para mañana traer ejemplos de procesos e hilos, centralizados, uniprocesamiento. Ejemplo programado**

**Trabaja cierto aspecto generándolo con fork**

**Ejemplo en java, sin hilos, proceso unico**

**ALGORITMOS DE SINCRONIZACION**

**Algoritmo de Lamport - FERNANDO**

Se usa cuando los relojes físicos no pueden sincronizarse perfectamente, en lugar de usar tiempos absolutos, usa relojes lógicos para garantizar que si un evento A ocurre antes que un evento B, todos los nodos respetan el orden

Numero mas grande = Quien ha realizado el último movimiento

VENTAJAS

Se garantiza un orden lógico sin necesidad de sincronizar relojes físicos

DESVENTAJA

No refleja el tiempo real, solo ordena los eventos

**Algoritmo de Cristian**

Se utiliza en sistemas distribuidos donde un cliente necesita sincronizar su reloj con un servidor de tiempo. Funciona bien cuando se asume que la latencia de la red es simétrica

DESVENTAJA

Si la latencia de la red es asimétrica, el cálculo del tiempo tendrá errores

**Algoritmo de Berkeley**

Este algoritmo permite que un nodo maestro determine un tiempo promedio y ajuste todos los relojes para minimizar

**Algoritmo de Bully - RUBEN**

TIPO DE SINCRONIZACION EN SISTEMAS DE VIDEOCONFERENCIAS

PRACTICA 4 DEBE TENER ALGUN ALGORIMTO DE SINCRONIZACION

OBJETOS DISTRIBUIDOS

Son objetos cuyas operaciones pueden ser invocadas de manera remota en una maquina diferente a la que reside el objeto

CORBA surge como propuesta para poder generar objetos distrbuidos con cualquier lenguaje

Fue el primero

RMI le gana a corba

Manejar objetos distribuidos

EJEMPLOS DE CORBA, DCOM,

DESCARGAR LIBREBIA DCOM Y CORRER OBJETO DISTRIBUIDO

**MIERCOLES**

**Establecer ejemplo de DCOM**

**Establecer ejemplo de CORBA funcionando y operando, tendrá buen porcentaje de participación en la materia**