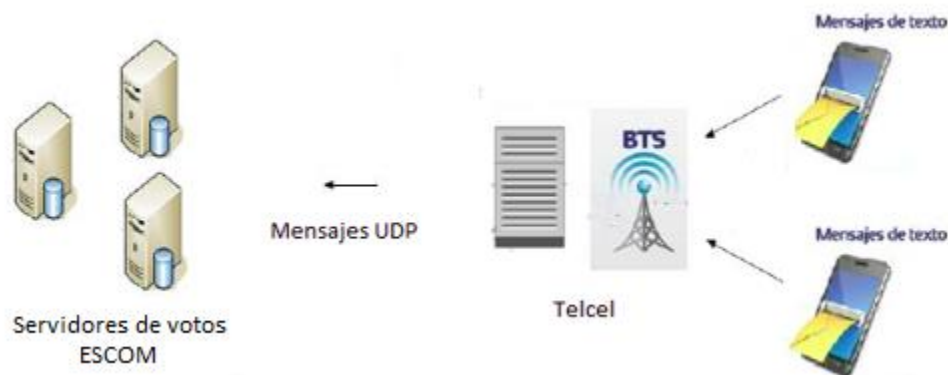


SDEP distribución de procesamiento

Elaborado por: Ukranio Coronilla

Como pudo observar en la práctica anterior, existen aplicaciones de cómputo que no pueden ser realizadas sólo con una computadora. En nuestro caso, una computadora no es suficiente para procesar los votos recibidos por setenta millones de votantes en el tiempo que duran las elecciones. Principalmente porque para determinar si se ha repetido un número telefónico cuando ya existen muchos votos, se incrementa el tiempo de respuesta, sin mencionar que la cantidad de memoria ocupada crece mucho. Cabe mencionar que podría haber más requerimientos, por ejemplo, validar que una CURP no se repita, lo cual lleva aún más recursos que solo validar el número de celular. También podría existir el requerimiento de validar que la CURP sea consistente con los registros que mantiene el INE en su base de datos, entre otros.

Dado que una computadora no es suficiente, ahora tendremos una aplicación distribuida que básicamente es como se muestra en la imagen:



Ejercicio 1

Construya la aplicación distribuida resumida en la imagen anterior retomando los códigos realizados en las practicas anteriores, teniendo presente en todo momento el mejor desempeño de la aplicación. A continuación, se describen las principales ideas que se deben implementar:

Para distribuir el procesamiento se sugiere hacer la siguiente división (aunque usted puede proponer otra al profesor antes de implementarlo si así lo desea), considerando que solo se tienen tres servidores de votos en su equipo. El servidor de votos A procesará los votos cuyo número telefónico termina con 0, 1, 2 y 3; el servidor de votos B procesará los votos cuyo número telefónico termina con 4, 5 y 6, mientras que el servidor de votos C procesará los votos cuyo número telefónico termina con 7, 8 y 9. Se sugiere que esta distribución se mejore asignando mayor rango a la computadora más rápida y menos carga a la más lenta. Optimice su sistema

asignando la proporción idónea (balanceo de carga), para lo cual debe comentar la idea de su equipo con el profesor antes de implementar.

Dado que estamos modelando la infraestructura de Telcel junto con los emisores de los mensajes de texto en una computadora cliente, y que tenemos tres servidores de votos, para distribuir los mensajes del cliente hacia los servidores tenemos dos opciones principales:

1 Que el cliente envíe los mensajes con último número de celular 0, 1, 2 y 3 al servidor A, los 4,5 y 6 al servidor B y los 7, 8 y 9 al servidor C.

2 Que el cliente envíe mensajes multicast idénticos a los tres servidores, y que cada servidor se encargue de procesar el que le corresponda.

Cada equipo decide cuál de las dos opciones le parece más conveniente utilizar de acuerdo a lo visto en teoría. Independientemente de la opción que se elija deben cumplirse los siguientes rubros:

- La comunicación unicast o multicast debe ser confiable de manera que se mantenga la consistencia aún con pérdida de mensajes. (Véase 8.4 Comunicación en grupo confiable - Tananbaum)
- Si utiliza unicast debe crear un hilo para enviar mensajes a cada servidor de manera que se utilice el protocolo solicitud respuesta de manera concurrente con los tres servidores y mejorar así el desempeño. De lo contrario el cliente tendría que esperar a que responda el servidor A para mandar solicitud al servidor B y esperar para mandar a C, lo cual implica un desperdicio de tiempo.