4.5. TOLERANCIA DE FALLAS

Decimos que un sistema falla cuando no cumple su especificación. En algunos casos, como en un sistema de ordenamiento distribuido en un supermercado, una falla podría provocar la falta de frijoles enlatados en una tienda. En otros casos, corno en un sistema distribuido para el control de tráfico aéreo, una falla podría ser catastrófica. Como las computadoras y los sistemas distribuidos se utilizan cada vez más en misiones donde la seguridad es crítica, la necesidad de evitar las fallas es cada vez mayor.

Fallas de componentes

Los sistemas de cómputo pueden fallar debido a una falla en algún componente, como procesador, la memoria, un dispositivo de E/S, un cable o el software. Una falla es un desperfecto, causado tal vez por un error de diseño, un error de fabricación, un error de programación, un daño físico, el deterioro con el curso de tiempo, condiciones ambientales adversas (pudo nevar sobre la computadora), entradas inesperadas, un error del operador, roedores comiendo parte del sistema y muchas otras causas. No todo esto conduce (de inmediato) a fallas del sistema, pero algunas de estas cosas sí.

Las fallas se clasifican por lo general como transitorias, intermitentes o permanentes. Las fallas transitorias ocurren una vez y después desaparecen. Si la operación se repite, la falla ya no se presenta. Si la transmisión expira y se repite, es probable que funcione la segunda vez, si ocurre una falla intermitente, ésta desaparece, reaparece, etcétera. Un mal contacto de un conector causa con frecuencia una falla intermitente, las cuales son graves por su difícil diagnóstico. Una falla permanente es aquella que continúa existiendo hasta reparar el componente con el desperfecto. Los circuitos quemados, los errores del software y el rompimiento de la cabeza del disco provocan con frecuencia fallas permanentes. El objetivo del diseño y construcción de sistemas tolerantes de fallas consiste en garantizar que el sistema continúe funcionando de manera correcta como un todo, incluso en la presencia de fallas.

Las fallas pueden ocurrir en todos los niveles: transistores, circuitos, tarjetas, procesadores, sistemas operativos, programas del usuario, etc.

Tolerancia de fallas mediante respaldo primario

La idea esencial del método de respaldo primario es que en cualquier instante, un servidor es el primario y realiza todo el trabajo. Si el primario falla, el respaldo ocupa su lugar. En forma ideal, el reemplazo debe ocurrir de manera limpia, y ser notado únicamente por el sistema operativo cliente, no por los programas de aplicación.
La tolerancia de fallas con respaldo primario tiene dos ventajas principales sobre la réplica activa. En primer lugar, es más sencilla durante la operación normal, puesto que los mensajes van sólo a un servidor (el primario) y no a todo un grupo. Los problemas asociados con el ordenamiento de estos mensajes también desaparecen. En segundo lugar, en la práctica se requieren menos máquinas, puesto que en cualquier instante se necesitan un primario y un respaldo (aunque cuando un respaldo se pone en servicio como primario, se necesita un nuevo respaldo de manera instantánea). Como desventaja, trabaja mal en presencia de fallas bizantinas, en las que el primario afirma erróneamente que funciona de manera perfecta. Además, la recuperación de una falla del primario puede ser compleja y consumir mucho tiempo.

Como ejemplo de la solución con respaldo primario, consideremos el protocolo simple de la figura 4-22, en donde se muestra una operación de escritura. El cliente envía un mensaje al primario, quien realiza el trabajo y después envía un mensaje de actualización al respaldo. Cuando el respaldo lo recibe, realiza el trabajo y entonces envía un reconocimiento de regreso al primario. Cuando llega el reconocimiento, el primario envía la respuesta al cliente.

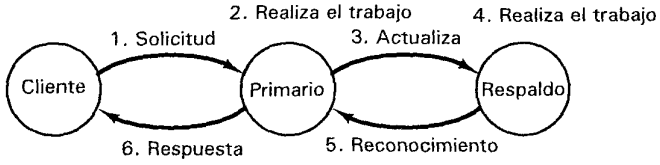

Figura 4-22. Un protocolo simple de respaldo primario en una operación de escritura.

Consideremos ahora el efecto de una falla del primario durante varios momentos durante una RPC. Si el primario falla antes de realizar el trabajo (paso 2), no hay ningún daño. El cliente expira y vuelve a intentar. Si intenta las veces suficientes, entonces de manera eventual llegará al respaldo y el trabajo se realizará con exactitud. Si el primario falla después de realizar el trabajo pero antes de enviar la actualización, cuando el respaldo ocupa su lugar y la solicitud vuelve a llegar, el trabajo se realizará por segunda vez. Si el trabajo tiene efectos colaterales, esto podría ser un problema. ‘Si el primario falla antes del paso 4 pero antes del paso 6, el trabajo podría terminar realizándose tres veces, una por el primario, otra por el respaldo como consecuencia del paso 3 y otra después de que el respaldo se convierte en el primario. Si solicita identificadores de acarreo, entonces podríamos garantizar que el trabajo sólo se realiza dos veces, pero hacer que se realice una vez es difícil o imposible.