Reintegração de Servidores em um Sistema de Replicação de Arquivos

Marcia Pasin (pasin@inf.ufrgs.br)
Taisy Silva Weber (taisy@inf.ufrgs.br)

Curso de Pós-Graduação em Ciência da Computação
Universidade Federal do Rio Grande do Sul
Caixa Postal 15064 CEP 91501-970

Resumo

Sistemas distribuídos representam uma plataforma ideal para implementação de sistemas computacionais com alta confiabilidade e disponibilidade devido a redundância fornecida por um grande número de estações interligadas. Falhas em uma estação servidora podem ser contornadas pela reconfiguração do sistema. Entretanto, falhas em seqüência que afetem múltiplas estações comprometem não apenas o desempenho do sistema, mas também a continuidade do serviço e sua confiabilidade. Servidores falhos, que tenham sido isolados do sistema, devem ser reintegrados tão logo quanto possível. Este artigo trata de sistemas de arquivos replicados e da reintegração de servidores nestes sistemas. É assumido um ambiente distribuído que garante alta confiabilidade em aplicações convencionais através da técnica de replicação de arquivos.

Abstract

Distributed systems are an ideal platform to develop high reliable computer applications due to the redundancy supplied by a great number of interconnected workstations. Failed stations can be masked reconfiguring the system. However, sequential faults, that affect multiple stations, not just decrease the performance of the system, but also affect the continuity of the service and its reliability. Thus, failed stations working as servers, that have been isolated from the system, should be reintegrated as soon as possible. This work is about replicated file systems and reintegration of failed servers in this systems. It is assumed a distributed environment that guarantees high reliability in conventional applications through replication of files.

1. Introdução

A replicação de arquivos é uma técnica de tolerância a falhas utilizada para aumentar a disponibilidade e a confiabilidade em sistemas distribuídos. Esta técnica dissemina *cópias de arquivos* entre várias estações servidoras em um sistema. Assim, se uma cópia é perdida acidentalmente, há outra para substituí-la.

Uma abordagem de distribuição destas cópias é a *cópia primária* [BUD 93] Nesta abordagem centralizada, um dos servidores é responsável pela coordenação dos demais servidores. O coordenador contém a cópia primária (e é chamado *servidor primário*) e os demais são *backups* (ou *servidores secundários*). Cada cliente sabe qual servidor é o primário e estabelece comunicação somente com este servidor. Os servidores secundários são apenas repositórios de dados.

Na escrita, o cliente envia o arquivo ao servidor primário. O servidor primário atualiza o arquivo em seu sistema de arquivos e envia uma cópia do arquivo para cada um dos servidores secundários por *difusão de escritas*. Para realizar a leitura de um arquivo, um cliente faz uma

requisição ao servidor primário. O servidor primário realiza a leitura e retorna a informação para o cliente.

Outra abordagem de distribuição é as *réplicas* ou *cópias ativas* [SCH 90]. Esse método submete todas as réplicas às mesmas regras. O controle da replicação não é centralizado como no método de cópia primária. No procedimento de escrita, a invocação do cliente é recebida por todas as réplicas. Cada réplica processa a alteração e retorna a resposta ao cliente. O cliente espera até receber a primeira resposta ou a maioria de respostas idênticas. Na leitura, o cliente faz a invocação e, novamente, espera até que receba a resposta.

O método das réplicas ativas requer que as cópias livres de falhas recebam as invocações dos clientes na mesma ordem. Isso pode ser resolvido através de uma primitiva de comunicação que satisfaça as propriedades de ordenação e de atomicidade.

2. Modelo do Sistema Considerado

Neste artigo considera-se um sistema replicado (por cópia primária ou réplicas ativas) composto por uma rede de estações com poucos servidores e muitos clientes. Os servidores que contém réplicas de um mesmo arquivo formam um *grupo de replicação*. Um *grupo de replicação* fornece os serviços básicos de leitura e escrita para os clientes. Quando um servidor do grupo de replicação falha, os servidores operacionais precisam garantir que a informação do sistema não foi comprometida (isto é, os arquivos indisponíveis possuem cópias no sistema para substituí-los) e continuar o serviço para os clientes. Após a detecção da falha, o servidor deve ser reparado. Então, é necessário reintegrar o servidor à rede.

O sistema considerado tolera apenas um ponto de falha de *crash* por vez em servidor, isto é, apenas um dos *n* servidores pode falhar em dado instante. Quando uma falha é detectada, o servidor deve ser confinado, reparado e reintegrado ao grupo para não comprometer a capacidade de tolerância a falhas do sistema. Apenas falhas em servidores serão consideradas, pois falhas em servidor comprometem todas as estações as quais este servidor presta serviço. Falhas em cliente comprometem apenas uma estação (o próprio cliente).

3. Fases de Operação do Sistema Distribuído Replicado

Pode-se distinguir duas fases na operação de um sistema distribuído com vários servidores: *operação plena* e *operação degradada*; além de duas operações básicas que podem ser realizadas com servidores sob este sistema durante sua vida útil: *reintegração de servidor* e *confinamento de servidor*.

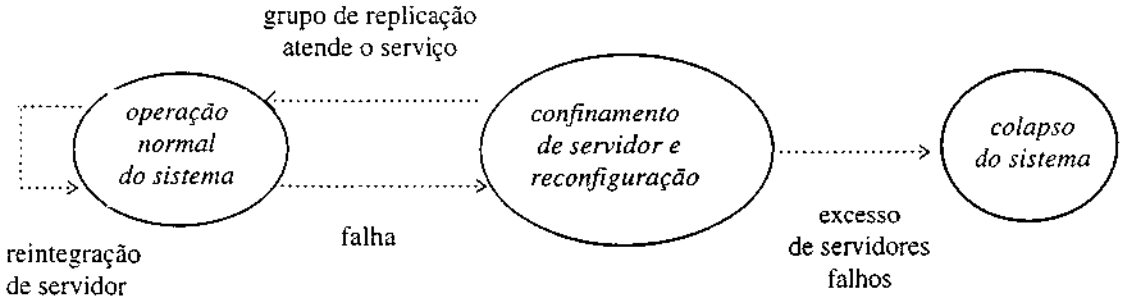

Figura 1 - Fases de operação do sistema distribuído

Durante a *fase normal* (plena ou degradada) (fig. 1), quando uma falha é detectada em um servidor, a estação deve ser *confinada*, tornando seu serviço inacessível. O sistema perde um

servidor e precisa de mecanismos para restaurar o serviço, *reconfigurando* o grupo de replicação. O sistema pode requisitar a substituição ou reparo do servidor perdido.

Depois do *confinamento* do servidor falho e da reconfiguração dos servidores, o sistema retorna a *fase normal* (degradada). Porém, a capacidade de tolerar falhas é reduzida. Um sistema robusto deve suportar determinado número de falhas subsequentes. Esta capacidade está associada ao número de cópias de arquivos disponíveis em cada grupo de replicação.

Para aumentar ou manter a capacidade de um sistema tolerar falhas, o sistema pode sofrer a *reintegração de servidor*. A reintegração começa quando um servidor é integrado fisicamente à rede. Envolve a *atualização do sistema de arquivos do servidor* e a *integração deste com o grupo de replicação*. Para realizar a atualização de suas cópias, o servidor troca mensagens com o grupo de replicação para obter a versão mais recente dos arquivos. Quando o protocolo de atualização termina, o grupo é notificado que o servidor está atualizado e que deverá voltar a participar ativamente das operações do sistema.

4. Reintegração de Servidores

A reintegração de servidores não é assunto facilmente encontrado na literatura. Na maioria das vezes trata-se de um procedimento manual, mas é necessária para prolongar a vida útil do sistema: (a) mantém a atividade plena do sistema, corrigindo uma eventual degradação de desempenho gerada por falha; (b) mantém o número de servidores da configuração original do sistema e (c) aumenta a confiabilidade e disponibilidade de informação, quando um novo servidor é adicionado.

A reintegração de servidor começa quando o servidor é ligado e difunde uma requisição de reintegração para o grupo de replicação ao qual quer se conectar. O grupo responde à requisição e começa o *protocolo de atualização do sistema de arquivo do servidor* (fig. 2).

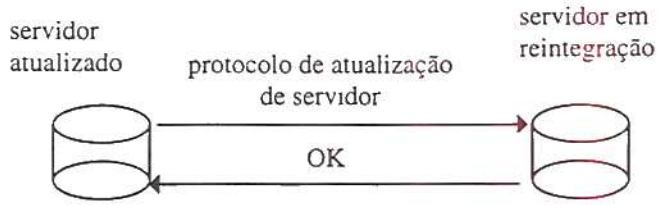

Figura 2 - Protocolo de atualização de servidor

O servidor em reintegração precisa coletar informação do grupo de replicação ao qual quer se integrar para atualizar o seu sistema de arquivos. Quando a atualização terminar, o grupo de replicação deve ser notificado em um procedimento final. São considerados três diferentes protocolos de *atualização de servidor* [LEB 96]: *transferência de volume*[1], *cópia de arquivos* e *retenção de logs*.

A *transferência de volume* é o protocolo mais simples. A partir de dois servidores que decidam que um volume deve ser recuperado, simplesmente o volume atual é copiado completamente para o servidor desatualizado. Um algoritmo de cópia recursiva percorre a árvore de diretórios de um servidor atualizado e transfere todos os arquivos para o servidor desatualizado. Este protocolo é indicado quando um novo servidor é inserido na rede e precisa atualizar todo o seu sistema de arquivos.

[1] conjunto de diretórios de arquivos de um servidor

A *cópia de arquivos* é um protocolo mais restrito que o anterior. Apenas são atualizados os arquivos alterados quando o servidor esteve ausente. Este protocolo é indicado para servidores que ficaram temporariamente desligados. A atualização pode ser realizada com a implementação de um algoritmo recursivo semelhante à transferência de volume, mas que utilize algum critério de comparação, ao invés de realizar a cópia incondicional. O critério da comparação pode ser números de versão associados às cópias dos arquivos. O número de versão inicialmente recebe o valor zero (criação do arquivo ou diretório) e é incrementado após cada operação de escrita realizada.

A *retenção de logs* aloca um espaço no disco de um servidor para servir como *cache* de operações para recuperar servidores falhos. Sempre que uma operação solicitada por um cliente não for transmitida para um servidor em decorrência de falha, será anotada na *cache*. Quando a falha no servidor for corrigida, o servidor com o *log* passa a retransmitir ao servidor que sofreu a falha todas as operações perdidas, tornando-o atualizado. Devido a necessidade de manutenção do *log*, este protocolo deverá ser utilizado para tratar falhas de curta duração.

5. Tornando o NFS Mais Confiável

O NFS (*Network File System*) [SAN 85] é o sistema de arquivos cliente-servidor mais conhecido desde sua implementação na década de 80. A versão atual do NFS comporta muitas necessidades que foram surgindo através de sua ampla utilização, mas ainda não é um sistema completo quando enfocamos alta disponibilidade do serviço. Alta disponibilidade é um determinante crítico, principalmente, para sistemas financeiros.

O protocolo de reintegração de servidor, bem como os métodos de replicação por *software* e *hardware*, podem ser aplicados ao NFS convencional para obter confiabilidade e disponibilidade desejadas. Um exemplo desta tecnologia é o RNFS (*Reliable Network File System*) [LEB 96, LEB 98] que está em desenvolvimento. No RNFS, o NFS convencional foi estendido para suportar replicação de arquivos por *cópia primária* sem utilizar *hardware* especial. O projeto do RNFS prevê a reintegração de servidores ao grupo de replicação. Um protótipo para a reintegração de servidores RNFS está sendo implementado.

6. Prototipação da Reintegração de Servidores

Um protótipo para a atualização de servidores durante a reintegração foi implementado para o RNFS. O protótipo usa o *rpcgen*, que permite a construção de aplicação distribuída com RPC (*Remote Procedure Call*). O cliente da aplicação (servidor secundário que está sendo reintegrado) faz as requisições para o servidor da aplicação (servidor primário). O RNFS assume que o servidor primário conterá sempre uma cópia atualizada de todos arquivos replicados que gerenciar [LEB 96]. Se o primário falhar, um novo primário já foi escolhido dentro do grupo de replicação antes do início da reintegração. Assim, o protótipo considera o servidor primário corrente como fonte de dados incondicional para qualquer protocolo de reintegração.

A parte pronta do protótipo implementa a atualização por *transferência de volume* e por *cópia de arquivos*. Na implementação, o sistema de arquivos de um servidor qualquer é atualizado a partir da informação do servidor primário. A atualização por transferência de volume foi implementada usando as RPCs disponíveis no NFS convencional.

Para implementar a atualização por cópia de arquivos foi necessária a inclusão do número de versão associado a cada arquivo do sistema. Antes de realizar a atualização do arquivo no servidor, a versão do arquivo no primário é consultada. Se a versão do primário para o arquivo

é maior que a versão do servidor que está sendo reintegrado, o arquivo é atualizado. Toda a comunicação adicional entre os servidores envolvidos na atualização é realizada por RPC.

7. Conclusões

Sistemas [BHI 91, LIS 91, LEB 98] foram propostos para tornar o NFS mais confiável. Esses sistemas utilizam replicação de *hardware* [BHI 91] ou *software* [LIS 91, LEB 98] e possuem preocupação especial em prover um ambiente altamente confiável e disponível com detecção automática de falhas, sem alterar a visão convencional do NFS para o usuário. Detectar a falha e confinar o servidor não é suficiente para manter a confiabilidade e disponibilidade esperadas para um sistema. O ideal é que estes sistemas permitam a reintegração automática de servidor.

Um protótipo com os protocolos de atualização foi implementado para realizar a reintegração automática de servidor em um sistema replicado baseado no NFS. Os protocolos usam as RPCs do NFS convencional, além de uma RPC que retorna o número de versão do arquivo para o protocolo de atualização por cópia de arquivos. Para eliminar a necessidade de usar essa RPC adicional, e controlar a desvantagem de tornar o protótipo incompatível com o NFS, uma solução usando a informação que indica a última atualização do arquivo pode ser tentada, se o sistema suportar um algoritmo de sincronização de *clocks*.

Os resultados obtidos pelo protótipo mostraram que, para o mesmo sistema de arquivos, a atualização por cópia de arquivos é mais eficiente que a transferência de volume, exceto se o servidor precisa atualizar todos ou a maioria dos arquivos em seu sistema. Neste caso, o tempo computado para a cópia de arquivos engloba a transferência do volume completo e todas as comparações necessárias para verificar que todo o volume está desatualizado.

Bibliografia

[BHI 91] BHIDE, A., ELNOZAHY, E. N., MORGAN, S. P., A highly available network file server. In: USENIX, 1991. **Proceedings...** [S.l.: s.n.], 1991. p.199-205.

[BUD 93] BUDHIRAJA, N., MARZULLO, K., SCHNEIDER, F., B., TOUEG, S. The primary-backup approach. In: MULLENDER, Sape (Ed.). **Distributed Systems**. 2 ed. New York: ACM Press, 1993. p.199-216.

[LEB 96] LEBOUTE, Mario M. **RNFS - Um sistema de arquivos distribuídos tolerante a falhas para o UNIX**. Porto Alegre: CPGCC da UFRGS, 1996. 85p.

[LEB 98] LEBOUTE, M., WEBER, Taisy S., A reliable distributed file system for UNIX based on NFS. In: IFIP INTERNATIONAL WORKSHOP ON DEPENDABLE COMPUTING AND ITS APPLICATIONS, 1998. **Proceedings...** Johannesburg, 1998. p.158-168.

[LIS 91] LISTOV, Barbara *et al.* Replication in the Harp File System. **Operating System Review**, New York, 1991. v.25, n.5. p.26-238.

[SAN 85] SANDBERG, R., GOLDBERG, D., KLEIMAN, S., WALSH, D., LYON, B. Design and implementation of the Sun Network File System. In: THE SUMMER USENIX CONFERENCE, 1985. **Proceedings...** [S.l.: s.n.], 1985. p.119-130.

[SCH 90] SCHNEIDER, F. B. Implementing Fault-Tolerant Services Using the State Machine Approach: A Tutorial **ACM Computing Surveys,** New York, v.22, n.4, December 1990.