2
Trabalhos Relacionados

Nesse capítulo, apresentamos os trabalhos relacionados ao GridFS, entrando em mais detalhes sobre os sistemas citados durante a introdução e realizando algumas considerações sobre a aplicação dos sistemas em grades e ambientes heterogêneos.

O protocolo FTP, através de servidores e clientes, pode ser usado para permitir o compartilhamento de arquivos. Os servidores FTP devem exportar uma árvore de diretórios e permitir que os clientes acessem os dados exportados. Em vez de permitir o acesso remoto, servidores FTP são usados para realizar cópias dos dados entre uma fonte e um destino. Assim, o arquivo pode ser manipulado localmente pelos clientes. O FTP é um protocolo bem estabelecido e vários implementações estão disponíveis para diversas plataformas. Vários sistemas para grades computacionais têm se baseado em servidores FTP, ou em extensões destes, para oferecer o compartilhamento de arquivos entre os computadores da grade [1, 3].

Outros sistemas definem mecanismos de transferência alternativos, usando protocolos de comunicação específicos e, em alguns cenários, o problema de compartilhamento de arquivos também pode ser tratado pelo uso de sistemas de arquivos distribuídos tradicionais. A seguir, apresentaremos alguns trabalhos encontrados na literatura.

2.1
GridFTP

O GridFTP define extensões ao protocolo FTP permitindo a transferência de dados de uma forma segura, confiável e com um alto desempenho. Uma implementação do serviço [1] está disponível no Globus Toolkit 4 [16].

O FTP foi escolhido como base para o sistema por várias razões: 1) pela existência de canais de controle separados dos canais de dados; 2) pela ampla difusão do protocolo; e 3) pela capacidade de definição de um conjunto de extensões previsto na sua especificação.

As principais características do GridFTP são:

- Controle de Transferência Externo. O cliente é capaz de iniciar e controlar a transferência de dados entre dois servidores remotos.
- Autenticação, Integridade e Confidencialidade. As diversas partes envolvidas na comunicação podem ser validadas com o uso do *Generic Security Service* no Globus. A definição de vários níveis de confidencialidade e verificação de integridade dos dados também é possível.
- Transferência Particionada (*striped*). Um conjunto de informações disponíveis em um conjunto de máquinas pode ser enviado para o destino em vários canais. Essa técnica visa fazer um uso otimizado da rede, onde, mesmo que uma máquina não consiga utilizar toda a capacidade da rede, um conjunto de máquinas faz o uso compartilhado resultando em um ganho de desempenho na operação.
- Transferência Paralela. Utiliza vários canais TCP paralelos entre a mesma fonte e destino, de forma a obter uma taxa agregada de transferência maior que a taxa de um único canal. Essa técnica pode ser usada juntamente a técnica anterior.
- Transmissão de Dados Parciais. Além da função de *resume* existente no FTP, o GridFTP também permite a transferência de regiões arbitrárias do arquivo, permitindo que apenas um determinado conjunto de informações seja transferido.
- Negociação automática do *buffer* TCP. O uso de um tamanho de *buffer* adequado pode aumentar significativamente o desempenho da comunicação em redes de longa distância. O GridFTP possui um mecanismo automático para escolha do tamanho do *buffer* e também permite que o valor seja atribuído manualmente.

Alguns resultados experimentais estão disponíveis em [1]. Os testes foram executados em três configurações de rede: LAN, a 612Mbps; MAN, a 1Gbps; e a WAN, a 30Gbps. Os testes comparam o GridFTP com outras implementações do protocolo FTP, tais como ncftp[1] e wuftp[2], e verificam

[1] http://www.ncftp.com/
[2] http://www.wu-ftpd.org/

o desempenho do sistema ao utilizar as extensões criadas. Na rede local, o GridFTP teve um desempenho aproximadamente 20% superior em relação às duas implementações utilizadas do protocolo FTP, e para a rede WAN o desempenho do GridFTP foi 15% superior ao wuftp e 30% superior ao ncftp. Ferramentas para medição do desempenho de rede e disco, como o Iperf[3] e Bonnie[4], foram utilizadas para determinar o limite que a transferência de dados poderia atingir. O GridFTP atingiu um desempenho próximo ao limite imposto pela rede e velocidade do disco.

2.2
Reliable File Transfer (RFT) Service

Embora o GridFTP ofereça suporte para a recuperação de falhas, o controle da operação é feito pelo cliente que solicita a transferência e uma falha nesse controle pode resultar em uma operação parcialmente concluída entre os servidores. Caso o cliente não possua um mecanismo para a retomada da operação, os dados permanecerão inconsistentes e a operação terá falhado. O canal de controle deve permanecer aberto durante toda a transferência, o que dificulta clientes móveis ou com uma conectividade instável. Nesse sentido, a criação de um serviço de transferência de arquivos de mais alto nível foi idealizada, onde clientes realizam a solicitação e o servidor é responsável por gerenciar a tranferência. O *Reliable File Transfer Service* [3, 17] pode ser utilizado para representar o cliente, permitindo que os usuários realizem solicitações de transferência através de descrições do trabalho e deixem a responsabilidade do controle da operação para o RFT. O sistema visa tratar a recuperação de falhas nos níveis de aplicação, rede e sistema e é responsável por controlar as transferências, mapeando as solicitações dos clientes em comandos a serem executados pelo GridFTP.

2.3
Global Access to Secondary Storage (GASS) Service

O GASS [18] define um espaço de nomes global através de URLs e, através da cópia dos dados remotos para uma área local, permite que as aplicações

[3] http://dast.nlanr.net/Projects/Iperf/
[4] http://www.textuality.com/bonnie/

acessem os dados fazendo uso das interfaces de entrada e saída oferecidas pelo sistema operacional. O objetivo do GASS consiste em disponibilizar um conjunto de funcionalidades para os sistemas de computação em grade, como transferência de executáveis e leitura de arquivos de configuração. O GASS considera os sequintes modos para acesso aos dados: 1) acesso somente leitura; 2) escrita compartilhada, sem controle de concorrência; 3) acesso para concatenação de informações, como em arquivos de log; e 4) acesso aleatório de leitura e escrita, também sem controle de concorrência. A abertura e o fechamento dos arquivos são feitos através de funções especiais, como, `globus_gass_fopen` e `globus_gass_fclose`. Essas funções ativam o mecanismo de *cache* do GASS e realizam a cópia do arquivo remoto de acordo com o estado do sistema. As aplicações que necessitam do acesso aos dados devem ser alteradas para realizar as chamadas específicas do GASS.

2.4 Stork

O Stork [19] foi desenvolvido com o objetivo de tratar as transferências de dados na grade de uma forma sistemática, assim como são tratados os processos submetidos pelos usuários. As operações de transferências são consideradas de uma forma tão importante quanto a submissão de processos. O Stork é um agendador das operações de transferências de dados na grade, permitindo que essas operações sejam agendadas, executadas, monitoradas, gerenciadas e retomadas em caso de falha, garantindo uma forma confiável para a realização de cópias de arquivos entre as máquinas. Os principais desafios enfrentados no sistema se referem à heterogeneidade dos recursos, ao tratamento das falhas nas operações, aos diferentes requisitos de transferência (em relação as necessidades das aplicações), e às limitações dos recursos a serem utilizados. Além disso, as falhas no Stork podem ser tratadas pela utilização de diversos protocolos de transferência diferentes, na tentativa de minimizar os erros irrecuperáveis ao tentar outros protocolos caso algum deles falhe. O uso compartilhado de CPU e da rede [20] pode resultar em ganhos gerais de desempenho, mantendo uma alta utilização de ambos os recursos ao longo do tempo, dado que eles podem ser usados independentemente.

2.5 DiskRouter

O DiskRouter [21] visa otimizar a transferência de dados em redes de longa distância fazendo com que a informação seja tratada de forma especializada em cada ponto da rede. Ao utilizar *buffers* em disco e memória em estações intermediárias entre a origem e o destino, o desempenho geral da transferência pode ser melhorado. O DiskRouter também pode ser usado como um ponto de armazenamento de dados próximo aos centros de processamento. Dessa forma, o resultado de algum processo pode ser rapidamente transferido para uma estação próxima, liberando a máquina para o processamento de outra tarefa. Em seguida, o DiskRouter se responsabiliza por transferir o resultado para o destino através da rede de longa distância. Além disso, é possível fazer uma avaliação dos dados em trânsito permitindo que o sistema se adapte em relação aos protocolos utilizados. O DiskRouter pode ser usado conjuntamente com o Stork, definindo estratégias para o fluxo dos dados.

2.6 Avaki Data Grid

O Avaki Data Grid [22] usa uma abordagem baseada em uma federação de servidores para disponibilizar diversos sistemas de arquivos de uma forma integrada. Cada membro interessado em disponibilizar uma árvore de arquivos deve iniciar um servidor, chamado de *Share Server*, que servirá como fonte de dados para o *Data Grid Access Server* (DGAS). Os DGAS atuam como um servidor NFS que pode ser montado na máquina cliente, logo, ao montar um DGAS no sistema de arquivos local, as aplicações têm acesso transparente aos dados armazenados no Avaki. O sistema é implementado em Java e uma parcela de código nativo é utilizada para permitir operações que a biblioteca padrão de Java não oferece.

2.7 Sistemas de Arquivos Distribuídos

Os sistemas de arquivos distribuídos tradicionais, como o NFS [8] e AFS [9], permitem que as aplicações acessem os dados como se eles fossem

locais. No NFS, através da definição de *mount points* para um servidor de arquivos remoto, a aplicação observa os dados no sistema como se fossem arquivos regulares no disco.

Um uso típico do NFS em uma rede local consiste na configuração de um servidor e vários clientes. Dessa forma, todas a máquinas que possuem um cliente são capazes de acessar os arquivos armazenados no servidor remoto. O NFS assume que as identidades dos usuários são compartilhadas na rede onde o servidor e cliente estão executando, ou seja, que os usuários possuem o mesmo UID e GID em ambas as máquinas.

O AFS cria sistema de arquivos virtual, composto por células que representam os diversos sistemas de arquivos em múltiplas instituições. As células formam um sistema único, amplamente acessível, e um mecanismo de segurança baseado em listas de controle de acesso está disponível para validar os usuários e garantir o acesso aos dados.

O Google possui um sistema de arquivos específico para o tratamento de grandes volumes de informação [5]. O sistema se baseia no uso de dispositivos de armazenamento comuns e é capaz de manipular eficientemente grandes arquivos, da ordem de gigabytes. Além das operações comuns dos sistemas de arquivos, como criação, deleção, abertura, fechamento, leitura e escrita, duas funções especiais foram definidas visando prover uma funcionalidade especial ao sistema: a função *append* é capaz de escrever uma série de bytes atomicamente e leva em consideração a presença de escritas concorrentes por vários clientes; enquanto a função *snapshot* é usada para criar cópias de uma sub árvore de forma instantânea, facilitando operações de *rollback* e ao mesmo tempo minimizando a interrupção das demais operações de escrita. O Google File System visa atender as necessidades do Google no armazenamento de dados.

2.8 Considerações sobre os sistemas

Em ambientes de computação em grade, o FTP pode ser usado como mecanismo de transferência. Porém, com o uso exclusivo do FTP, não é possível o acesso remoto aos dados e os arquivos devem ser transferidos completamente. Além disso, algum outro sistema de gerenciamento precisa ser utilizado para permitir um endereçamento dos arquivos e evitar o uso explícito de comandos

FTP por partes dos clientes.

Os serviços do Globus, especialmente o *Reliable File Transfer Service* e o *Global Access to Secondary Storage Service*, bem como o GridFTP, apresentam soluções para o compartilhamento de dados em grade. Porém, algumas funcionalidades importantes, como o cálculo de estimativas de transferência, e o suporte a metadados não estão disponíveis. O RFT e o GridFTP possuem um viés amplamente voltado para a transferência dos dados em si, enquanto o GASS, que define um espaço de nomes global, disponibiliza funções apenas para o acesso a dados remotos e força alterações nas aplicações para usar o sistema.

O Avaki Data Grid é o sistema que mais se assemelha aos requisitos apresentados no capítulo anterior. Porém, por ser uma solução comercial e proprietária, o Avaki não permite os benefícios do modelo de desenvolvimento de software aberto, onde os indivíduos podem inspecionar e aprimorar o software de acordo com suas necessidades. Além disso, os recursos de metadados e de estimativa do tempo de transferência entre os diversos nós não estão disponíveis no Avaki.

Um sistema como o GridFS, por se tratar de uma infra-estrutura básica para o gerenciamento de arquivos em rede, pode servir como base para a construção de diversos sistemas de mais alto nível. Por exemplo, podemos ter sistemas como o Stork e o DiskRouter implementados sobre o GridFS, permitindo que as aplicações de gerenciamento da grade, ou mesmo aplicações em outros contextos, movimentem e manipulem os arquivos armazenados para atingir o objetivo desejado.

No uso de sistemas de arquivos distribuídos tradicionais, ao considerar o aspecto de desempenho, a utilização de dados remotos pode degradar o funcionamento das aplicações que precisam acessar o arquivo de dados em múltiplas leituras. Nessas situações, é aconselhável que o arquivo seja replicado em uma área local sempre que possível. Por outro lado, caso o arquivo não precise ser acessado completamente, o acesso remoto pode ser mais eficiente do que a transferência integral do arquivo. É interessante permitir que a decisão sobre a realização da cópia do arquivo seja tomada de acordo com um estudo do padrão de uso de dados, de acordo com a aplicação.

O NFS, apesar de ser um sistema amplamente aplicado na indústria e centros de pesquisa, não oferece um suporte adequado aos ambientes de computação em grade, especialmente devido ao fato que ele foi modelado considerando padrões de uso e características de rede que não se aplicam aos

cenários atuais de computação em grade [22]. Ao considerar uma topologia de uso do NFS em que apenas um servidor é definido e vários clientes acessam os dados concorrentemente, a obtenção de um alto desempenho computacional é dificultada devido a uma sobrecarga do servidor central. Uma possível solução consiste na instalação de diversos servidores, e na configuração de clientes para cada servidor. Entretanto, visando conectar completamente o conjunto de máquinas, os administradores devem instalar n servidores e configurar n*(n-1) clientes, onde n é o número de máquinas do conjunto. Para evitar a configuração do cliente para cada um dos servidores, podemos montar todos os servidores NFS em um único local, e configurar os clientes para acessar essa raiz centralizada. Porém, essa abordagem irá gerar um alto tráfego nesse servidor, que servirá como ponte para os demais servidores. Além disso, ao considerar um cenário multi-institucional, o compartilhamento da base de dados de usuários, com os mesmos UID e GID, pode ser inviável.

O AFS resolve a maioria das questões citadas no NFS. Os principais problemas no uso do AFS em um ambiente de grade são: 1) ele requer que todos os parceiros utilizem o Kerberos como mecanismo de autenticação; e 2) dado que o AFS é estritamente um sistema de arquivos - e não um sistema sobreposto - todos os parceiros devem migrar os dados para ele. Garantir a homogeneidade nesses dois itens em um ambiente de várias instituições pode ser algo bastante difícil.

De acordo com as considerações apresentadas, podemos observar que os sistemas de arquivos distribuídos tradicionais, apesar de serem mais flexíveis do ponto de vista de formas de acesso e gerenciamento dos arquivos, apresentam algumas limitações para seu uso em grades computacionais. Dessa forma, as infra-estruturas especializadas para grades, notadamente os sistemas Globus e Condor, têm procurado oferecer soluções alternativas para o compartilhamento de arquivos.

A definição de sistemas especializados está presente em diversas áreas da computação. A avaliação dos requisitos de uma aplicação pode indicar a necessidade de um novo componente de software, desenvolvido com o objetivo de atacar um determinado problema de uma forma simples e objetiva. Como será visto no próximo capítulo, o GridFS foi projetado para atender a demanda específica do CSBase no compartilhamento de arquivos e foi idealizado de forma a combinar diversas características dos sistemas propostos pela comunidade. Apesar de possuir o CSBase como cliente direto, o GridFS pode ser usado em cenários de computação em grade de uma forma mais geral, ou em qualquer ambiente que necessite do compartilhamento de arquivos.