Análise Comparativa entre Soluções Livres para Construção de Sistemas de Arquivos Distribuídos

Pedro C. B. Azevedo Filho
SENAI/SC
Serviço Nacional de Aprendizagem Industrial
pedro.azevedo-filho@sc.senai.br

Douglas D. J. de Macedo
PPGEGC - dEGC
Universidade Federal de Santa Catarina
macedo@inf.ufsc.br

***Resumo* - Este estudo visa comparar o desempenho de algumas ferramentas para montagem de *clusters* para armazenamento em massa de dados. Estas soluções foram configuradas em cenários pré-estabelecidos e seu desempenho testado por uma solução de *benchmark* que avaliou a velocidade destas soluções ao realizar algumas operações. Após os testes foi possível realizar um comparativo do comportamento de cada uma das ferramentas em situações específicas.**

I. INTRODUÇÃO

Com o passar dos anos, a popularização dos computadores e serviços exigiu cada vez mais do desempenho e da disponibilidade dos serviços prestados. Após a década de 80, com o desenvolvimento de *hardware* e o avanço das tecnologias de comunicação [1] tornou-se possível a junção de computadores em elaboradas infraestruturas para melhoria do desempenho de algumas aplicações [2], conceito também conhecido por computação distribuída.

A computação distribuída possui várias vertentes de pesquisa e desenvolvimento, sendo que a necessidade de dados compartilhados e em conformidade neste tipo de sistema é uma realidade. Desta forma, muitas aplicações distribuídas utilizam como base um sistema para gerenciar a escrita e a leitura de dados [3], que deve funcionar de forma distribuída também.

Esta pesquisa se propõe a realizar uma análise comparativa entre algumas das soluções disponíveis para construção de sistemas de arquivos distribuídos. Cabe ressaltar que foram avaliadas somente ferramentas livres para plataformas GNU/Linux. Após a montagem das estruturas e realizados todos os testes necessários foram levantados dados para criar um comparativo das ferramentas e classificar conforme sua eficiência e desempenho em cada uma das situações propostas.

II. FERRAMENTAS RELACIONADAS

Neste trabalho, quatro soluções de sistemas de arquivos distribuídos foram avaliadas. Para a escolha destas, foi levado em consideração a estrutura e funcionamento, pois se fossem comparadas soluções totalmente diferentes e com o funcionamento diferente, torna-se extremamente complicado traçar uma taxonomia para os testes.

Das ferramentas escolhidas, duas delas são bastante conhecidas na área, possuem um amplo suporte, além da maturidade de anos de desenvolvimento, são elas: PVFS2 (Parallel Virtual File System) e o Lustre. As demais soluções trabalham de forma semelhante, porém ainda não são consideradas pela comunidade como soluções maduras, pelo baixo tempo de desenvolvimento, que é o caso do FhGFS (*Fraunhofer Parallel File System*) e do Ceph, ambas estão em desenvolvimento a cerca de cinco anos.

Desenvolvido na Universidade de Clemson nos Estados Unidos, o PVFS2 [6] é um sistema de arquivo distribuído, que visa o trabalho em ambientes de *clusters* computacionais. O projeto tem foco em ambientes com grande concorrência por I/O, inclusive com algumas implementações visando alta *performance* nestas estruturas, como é o caso do ROMIO. O PVFS2, também se caracteriza pela independência de hardware, o mesmo pode ser executado em diversas arquiteturas, além do amplo suporte a sistemas operacionais baseados no *kernel* Linux.

FranhoferFS ou FhGFS é um sistema de arquivos paralelo, desenvolvido no Instituto de Matemática Industrial (ITWM) na Alemanha. A primeira vista, nenhum sistema distribuído pode ser classificado como de fácil administração, mas impressiona a facilidade de lidar com o FhGFS, pela boa documentação e pela quantidade de *scripts* que deixam a tarefa menos difícil. O FhGFS permite também, o monitoramento da estrutura de forma interessante, repleto de gráficos mostrando o desempenho da solução e eventuais problemas, facilitando o trabalho de administrar o sistema.

O Ceph surgiu como um projeto de pesquisa de doutorado [4] na universidade da Califórnia nos Estados Unidos. Possui como característica forte a possibilidade de lidar com uma quantidade muito grande de informações, aliado a um forte desenvolvimento na área de replicação de dados e tolerância a falha. O *kernel* Linux (2.6.34) já trouxe suporte nativo ao Ceph de forma experimental, o que mostra o crescimento do projeto.

O Lustre [5] foi desenvolvido inicialmente por Peter Braam que era um cientista da Universidade Carnegie Mellon, o projeto tornou-se tão popular que Braam criou uma empresa, que foi adquirida pela Sun Microsystems. Hoje, o Lustre é desenvolvido e mantido pela Oracle. Em linhas gerais, o Lustre é considerado um sistema de arquivo muito versátil, escalável e estável, inclusive com alguns *cases* que chamam a atenção como o caso do supercomputador Plêiades da NASA que o utiliza como sistema de arquivos distribuído.

III. RESULTADOS EXPERIMENTAIS

A taxonomia dos testes levou em consideração a velocidade das soluções ao realizar algumas operações

básicas de um sistema de arquivos, como leitura, releitura, escrita e reescrita. As soluções foram configuradas em dois cenários distintos e efetuados os testes. Os testes foram realizados de forma simultânea a partir de dois clientes conectados a esta estrutura. Os testes de desempenho foram realizados utilizando uma suíte de testes chamada IOzone [7]. O IOzone, permite diversas configurações para realização dos testes e para a apresentação dos resultados.

A. Ambiente

Para que fosse possível analisar as ferramentas corretamente nos cenários definidos, foi necessário utilizar uma infraestrutura de redes específica. Nos dois cenários avaliados conta-se com seis computadores para a simulação tanto dos clientes como a dos servidores, sendo quatro servidores e dois clientes.

Os seis computadores possuem a mesma configuração referente a hardware, são máquinas com processador Pentium IV dual core de 3GHz, 3GB de memória RAM, HD de 80GB SATA II e com placas de rede FastEthernet (100Mb/s). Os computadores foram interligados via rede por meio de *switches* Cisco Catalyst 2950 de 24 portas, estes *switches* têm como característica todas as suas portas trabalhando a 100Mb/s (FastEthernet).

Em outra topologia conta-se, também, com a presença de dois roteadores Cisco 2600, estes roteadores foram configurados para simular o tráfego limitado e distribuído, ou seja, entre os dois roteadores foi configurado um link serial que pode atingir no máximo 2Mb/s de velocidade.

B. Cenários

A principal idéia por trás da definição de alguns cenários foi verificar o comportamento das ferramentas no momento dos testes em diferentes infraestruturas de rede. Um dos cenários mostra a implantação das soluções em um ambiente local e teoricamente sem problemas para que a comunicação ocorresse corretamente. Em outro cenário é verificado o desempenho das soluções em relação a uma infraestrutura distribuída, com seus objetos separados e com certas limitações de rede.

1) Cenário 1

O primeiro cenário para testes, mostrado na Figura 1, tem por intuído simular o ambiente de uma empresa com um local físico apenas, ou seja, todos os dados trafegando em uma rede local (LAN).

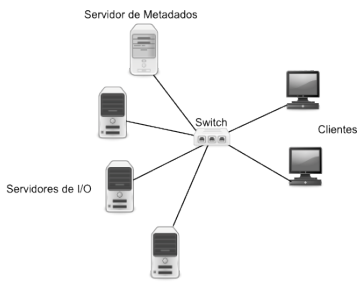

Figura 1. Cenário um para testes.

Este primeiro cenário conta com a configuração das soluções em três servidores de I/O, ou seja, os dados são armazenados em três servidores aliando as suas capacidades de armazenamento e um servidor de metadados. A manipulação dos dados armazenados pelo servidor ocorre por meio de dois computadores clientes. Todos os equipamentos, tanto servidores quanto clientes estão conectados a um *switch* FastEthernet, trabalhando a 100Mb/s (FastEthernet 100BASE-TX).

2) Cenário 2

O segundo cenário, mostrado na Figura 2, tem como principal objetivo, simular uma infraestrutura onde existem ambientes separados geograficamente e torna-se necessária a interligação destes dois ambientes por um link externo (geralmente operadora).

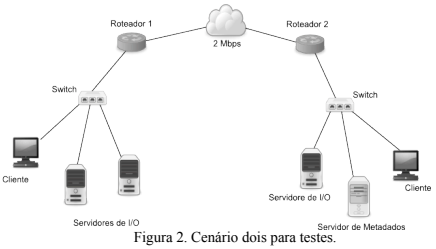

Figura 2. Cenário dois para testes.

Diferente do primeiro cenário apresentado, este possui como característica a distribuição dos dados em servidores geograficamente distantes, simulando o ambiente de uma empresa que possui mais de um espaço físico de trabalho e que necessita destes dados em conformidade.

Pode-se dividir este cenário em dois espaços físicos. No primeiro espaço físico temos um cliente conectado a mesma LAN de dois servidores de I/O. No outro espaço físico encontramos novamente a figura de um cliente e dois servidores conectados a mesma LAN, sendo que, um dos servidores é de metadados e o outro de I/O. Para permitir a comunicação entre os dois ambientes, foi utilizado um link serial (limitado a 2 Mb/s).

C. Resultados

Alguns pontos importantes devem ser levados em consideração para análise dos resultados. O primeiro deles diz respeito à instalação e configuração das ferramentas. Foram realizadas configurações padrões nas ferramentas, buscando avaliá-las sob a mesma ótica.

O segundo ponto importante faz referência à disposição dos equipamentos conforme os cenários apresentados. Cada um dos cenários apresenta características bem distintas e até mesmo em um mesmo cenário o desempenho de um cliente e outro pode ser bastante diferente. Um exemplo bastante claro disso é o cenário 2, onde de um lado encontramos um cliente próximo a dois servidores de I/O (responsáveis por gravar os dados) e o outro cliente próximo a somente um servidor de I/O.

Após tabular os resultados de todos os testes (cerca de 1600 testes - 25 amostras x 4 operações x 4 soluções x 2 clientes x 2 cenários), foram gerados gráficos para análise do desempenho de cada uma das ferramentas. Os gráficos foram gerados a partir da análise de cada uma das operações em cada um dos cenários propostos.

O gráfico possui como variáveis (eixos) a taxa de transferência realizada por cada uma das ferramentas (linhas) manipulando arquivos de tamanhos variados gerados, ou seja, o gráfico é apresentado mostrando a taxa de transferência (eixo Y) realizada para cada tamanho de arquivo (eixo X).

As unidades utilizadas nos eixos são expressas da seguinte forma: para expressar a taxa de transferência, a unidade de medida utilizada no eixo Y foi bytes por segundo. Já para expressar o tamanho dos arquivos manipulados apresentado no eixo X foi utilizada a unidade de bytes.

1) Operação diferente, desempenho diferente

O desempenho de algumas ferramentas varia muito, conforme a operação realizada. Isso demonstra o foco dado ao projeto da solução. A Figura 3 mostra o desempenho das ferramentas na operação de escrita e leitura no Cenário 1.

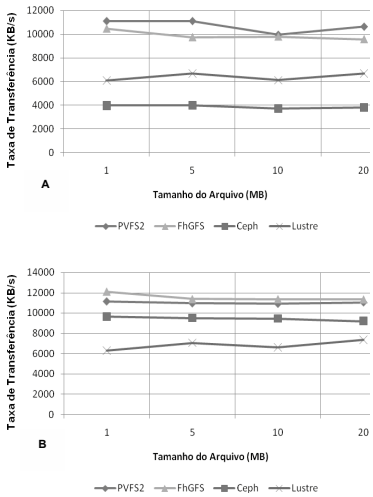

Figura 3. Desempenho das ferramentas na operação de escrita e leitura no cenário 1.

O primeiro ponto que chama a atenção é o salto de desempenho de algumas soluções, o Ceph, por exemplo, na operação de escrita (Figura 3a) apresentou desempenho médio de 4 MB/s, já quando é verificado o desempenho do mesmo na operação de leitura (gráfico 3b) há um salto para cerca de 10MB/s. O que pode ser facilmente verificado neste ponto, é que determinadas soluções dão mais foco ao trabalho com uma determinada operação ou uma aplicação específica. É muito importante avaliar estes dados no momento da escolha de uma solução.

2) Cenário diferente, desempenho diferente

O desempenho das soluções também varia muito de cenário para cenário. A Figura 4 mostra o desempenho das ferramentas na operação de reescrita no Cenário 1 e no Cenário 2.

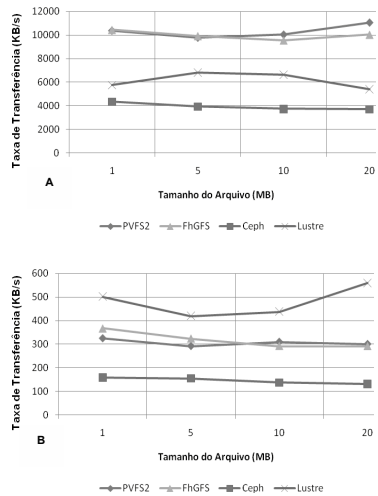

Figura 4. Desempenho das ferramentas na operação de reescrita no cenário 1 no cenário 2.

Neste ponto é verificada a diferença na implementação da parte de rede, pois, a identificação da rede e a facilidade de encontrar objetos mais próximos permitem uma melhora no trabalho com os recursos disponíveis na rede. O Lustre, por exemplo, mostrou um desempenho regular próximo aos 6 MB/s no cenário 1 (Figura 4a) onde a rede local toda trabalha a 100Mb/s, quando testado no cenário 2 (Figura 4b) o desempenho não foi tão bom assim, pois existe uma limitação grande da rede devido ao link de 2Mb/s dentro da infra-estrutura.

3) Clientes em redes diferentes, desempenho diferente

Um terceiro ponto importante de verificar, também atrelado as questões de cenário e até mesmo as operações, mostra o desempenho diferente entre clientes diferentes (mesmo havendo uma configuração de máquina igual, sistema operacional igual e etc). Principalmente no cenário 2, onde o cliente da rede 2 está próximo de somente um servidor de I/O. A Figura 5 mostra a variação do desempenho das ferramentas nos dois clientes do Cenário 2 realizando a operação de releitura.

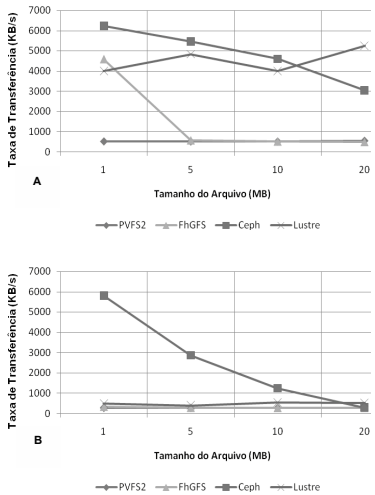

Figura 5. Desempenho das ferramentas na operação de releitura no cenário 2.

A importância de verificar os componentes da rede e também a preocupação do administrador, disponibilizar recursos para que a plataforma funcione de maneira adequada, auxilia e muito no desempenho das ferramentas. A grande diferença que existe no resultado apresentado na Figura 5a e na Figura 5b é que o cliente do primeiro gráfico está mais próximo de um servidor de I/O a mais, ou seja, tem mais recursos disponíveis.

4) Desempenho médio geral

Abaixo são apresentadas as médias de desempenho de cada uma das ferramentas, separadas em duas tabelas. Nas Tabelas 1 e 2 pode-se visualizar uma média do desempenho dos clientes no Cenário 1 e 2, respectivamente.

Solução	Escrita	Reescrita	Leitura	Releitura
PVFS2	10,42MB/s	10,44MB/s	11,02MB/s	11,12MB/s
FhGFS	10,01MB/s	9,98MB/s	11,52MB/s	11,51MB/s
Ceph	3,80MB/s	3,81MB/s	9,49MB/s	10,17MB/s
Lustre	6,31MB/s	6,48MB/s	6,49MB/s	6,79MB/s

Tabela 1. Desempenho médio das ferramentas no cenário 1

Solução	Escrita	Reescrita	Leitura	Releitura
PVFS2	321,04KB/s	339,1KB/s	382,93KB/s	397,92KB/s
FhGFS	816,15KB/s	851,37KB/s	915,28KB/s	921,94KB/s
Ceph	336,02KB/s	373,11KB/s	3,67MB/s	3,69MB/s
Lustre	2,0MB/s	2,14MB/s	2,44MB/s	2,5MB/s

Tabela 2. Desempenho médio das ferramentas no cenário 2

IV. CONCLUSÕES

Com todos os testes realizados, foi possível determinar o comportamento e o desempenho das ferramentas em cada uma das operações. Deve-se levar em consideração também, que todos os testes foram realizados em um ambiente de baixa escala e que talvez algumas das características aqui descritas não sejam identificadas em outras estruturas.

Depois da análise de todos os fatores referentes a desempenho e utilização das ferramentas, foi possível destacar pontos fortes e fracos das mesmas conforme cada uma das operações.

Dentre as ferramentas avaliadas, o FhGFS e PVFS2 foram as soluções que se destacaram no Cenário 1 em todas as operações. O desempenho destas duas aplicações em um cenário local aliado a sua facilidade de instalação, operação e manutenção, classifica muito bem estas duas.

No segundo cenário, a situação é bastante diferente daquela apresentada no primeiro, pois os testes neste cenário visaram verificar outras características das soluções, dentre elas, a replicação e capacidade de se adequar a rede. O Lustre apresentou um bom desempenho em todas as operações neste cenário, sendo que alcançou os melhores índices nas operações de escrita e reescrita. Já o Ceph apresentou bom desempenho quando tratando de operações que envolvam leitura.

Como sugestão de trabalhos futuros, pode-se estender a arquitetura proposta nos cenários e aplicá-los em diferentes softwares para sistemas de arquivos distribuídos, para efetuar uma comparação com a análise já realizada neste estudo. Ainda, é possível fazer os testes utilizando outros modelos de benchmark ou ainda, outros tipos de dados, buscando avaliar se isto afetará o desempenho dos sistemas de arquivos distribuídos.

REFERÊNCIAS

[1] Tanenbaum, Andrew S.; STEEN, Maarten Van. Sistemas distribuídos: princípios e paradigmas. 2. ed. São Paulo, SP: Pearson Prentice Hall, 2007. 1 p.

[2] Dantas, Mario. Computação distribuída de alto desempenho: redes, clusters e grids computacionais. Rio de Janeiro: Axcel Books, 2005. 278 p.

[3] Coulouris, George F.; Dollimore, Jean; Kindberg. Tim. Sistemas distribuídos: conceitos e projeto. 4. ed. Rio Grande do Sul: Bookman, 2007. 792p.

[4] Sage A. Weil, Scott A. Brandt, Ethan L. Miller, Darrell D. E. Long, and Carlos Maltzahn. 2006. Ceph: a scalable, high-performance distributed file system. In *Proceedings of the 7th symposium on Operating systems design and implementation* (OSDI '06). USENIX Association, Berkeley, CA, USA, 307-320.

[5] Paul Caspi, Jean-Louis Colaço Gérard, Marc Pouzet, and Pascal Raymond. 2009. Synchronous objects with scheduling policies: introducing safe shared memory in lustre. In *Proceedings of the 2009 ACM SIGPLAN/SIGBED conference on Languages, compilers, and tools for embedded systems* (LCTES '09). ACM, New York, NY, USA, 11-20.

[6] Philip H. Carns, Walter B. Ligon, III, Robert B. Ross, and Rajeev Thakur. 2000. PVFS: a parallel file system for linux clusters. In *Proceedings of the 4th annual Linux Showcase \& Conference - Volume 4* (ALS'00), Vol. 4. USENIX Association, Berkeley, CA, USA, 28-28.

[7] Norcott, W. IOZONE Filesystem Benchmark (2011), http://www.iozone.org.