

## Sistema de Arquivos Distribuídos: Avaliação dos Volumes do GlusterFS em relação aos tipos de *Workloads*

Fábio Leandro Rodrigues Cordeiro<sup>1</sup>, Carlos Augusto de Paiva da Silva Martins<sup>1</sup>

<sup>1</sup> Programa de Pós-Graduação em Engenharia Elétrica – PPGEE (PUC Minas)  
Caixa Postal 1.686 – 30.535-901 – Belo Horizonte – MG – Brasil  
{fabioleandro,capsm}@pucminas.br

**Abstract.** *The increasing use of computers and a large amount of data generated by several applications, make necessary the storage improvement of such data. This is a challenge when is considered a processors higher growth speed regarding input and output storage devices. Therefore, the Distributed File Systems are an alternative to minimize this problem. This paper proposes a review of the most common Gluster Filesystem volumes about the workloads that are generated by the IOzone benchmark tool. Thus, is identified the behavior of the GlusterFS volumes in environments with the submitted workloads.*

**Resumo.** *Com o crescimento da utilização de computadores e a gigante massa de dados gerada por diversas aplicações, aumenta a necessidade na melhoria do armazenamento desses dados. Isso é um desafio levando em consideração o maior crescimento da velocidade dos processadores em relação a velocidade de entrada e saída dos dispositivos de armazenamento. Neste contexto os Sistemas de Arquivos Distribuídos aparecem como alternativa para minimizar este problema. O presente trabalho propõe uma avaliação dos volumes mais comuns do Gluster Filesystem em relação à cargas geradas por meio da ferramenta de benchmark IOzone, identificando assim o comportamento dos volumes do GlusterFS nos ambientes propostos em relação às cargas submetidas.*

### 1. Introdução

Com o advento da Web e a crescente utilização de computadores em ambientes empresariais e em aplicações científicas, observa-se um aumento na geração de grande volume de dados. Consequentemente surge a necessidade de processamento e armazenamento em meios que permitam rápida recuperação e também disponibilidade desses dados. Em Noronha e Panda [1], sugere-se que este crescimento necessita de mecanismos que ajudem a diminuir a lacuna existente entre a velocidade dos processadores atuais, velocidade da rede e fundamentalmente a capacidade de E/S dos dispositivos de armazenamento.

Aplicações científicas como genética e meteorologia são exemplos práticos que necessitam de estratégias para aumentar o desempenho de armazenamento de dados. Tais áreas demandam organização e distribuição de carga, uma vez que os dados podem estar dispersos em vários computadores em rede. Fato que exige um alto poder de processamento objetivando uma melhoria no desempenho e exatidão do resultado esperado [2].

Nesse contexto, surgem os Sistemas de Arquivos Distribuídos (SADs), capazes de prover melhorias para esses dados. Conforme é mostrado em Mendez e Rexachs [3], as aplicações com finalidades científicas demandam por grande capacidade de processamento e armazenamento de dados que podem ser alcançadas pela utilização de SADs.

Outro fato que deve ser observado é a análise da carga de trabalho (*Workload*) à qual o SAD será submetido. É importante que sejam analisadas as características como tamanho, acesso, dentre outros comportamentos da carga em questão [4].

Também é importante observar que em ambientes de pequena escala, como pequenas empresas, existe uma quantidade de dados que, além da importância, podem ter um valor expressivo em volume de dados e necessitam de especificidades de segurança, como redundância e disponibilidade e desempenho. Em função destas demandas e com limitantes como alto valor de soluções robustas, a utilização de SADs de código aberto pode ser uma saída viável para atender estas demandas com baixo custo [2].

### **1.1. Problema**

Existe uma necessidade de identificação da melhor configuração do tipo de volume de um SAD, a ser utilizada em função das características de cada ambiente e das respectivas cargas de trabalho (*Workloads*).

Outra necessidade que levanta um problema a ser investigado é o comportamento destas cargas levando em consideração, além dos tipos de volumes, o tipo de conectividade de rede aos quais o SAD está sendo executado como ambientes *Fast Ethernet* e *Gigabit* que em geral são os cenários comuns em redes de pequeno porte.

### **1.2. Objetivos**

Esta pesquisa se propõe a avaliar e analisar um SAD executado nos ambientes *Fast Ethernet* e *Gigabit*, ambos controlados. Como estudo de caso, pretende-se utilizar o GlusterFS por meio da ferramenta de *benchmark* IOzone. Tendo em vista identificar seus comportamentos para tipos e tamanhos de cargas comuns às áreas científicas de pesquisa em universidades bem como dados comuns de ambientes de pequenas empresas.

O restante deste trabalho foi organizado com uma seção de fundamentação teórica, composta por contextualização de SADs e apresentação do GlusterFS. Depois uma seção com os trabalhos relacionados à esta pesquisa, outra de planejamento da avaliação com detalhes dos cenários avaliados e também, a classificação dos testes além da ferramenta para o *benchmark*. Seguida de uma seção de resultados e por fim a conclusão.

## **2. Fundamentação Teórica**

### **2.1. Sistemas de Arquivos Distribuídos (SADs)**

SADs foram desenvolvidos com objetivo de melhoria no processo de armazenamento em rede, através da realização de operações similares aos sistemas de arquivos locais em um sistema em rede, com a finalidade de propor melhorias dessas operações. O desenvolvimento de SADs possui registros desde as primeiras interconexões entre computadores, a ARPANET do departamento de Defesa dos Estados Unidos [5].

Um dos mais populares SADs é o Network File System (NFS), desenvolvido pela Sun Microsystems. Essa popularização se deu pela empresa ter disponibilizado seu conjunto de protocolos para que outras pudessem desenvolver clientes e servidores compatíveis. Por isso, atualmente podemos encontrar implementações para diversos SOs [5].

O NFS apresenta vantagens em relação a facilidade de implementação, embora alguns problemas como segurança e o alto consumo de rede além da dependência de um

servidor centralizado serem vistos como suas principais limitações. Até mesmo em sua versão melhorada, o NFS4, permite uma divisão entre alguns servidores, porém quando um deles fica indisponível, torna também inacessível dados que ali se encontram. Tais problemas citados, alinhados à necessidade de maior desempenho da computação concorrente, características intrínsecas de sistemas complexos como *Clusters*, trazem consigo a necessidade de adoção de SADs que possam suplantam tais requisitos [6].

Surtem então projetos para desenvolvimento de SADs, cada um destes trazendo características e conceitos que agregaram técnicas e conhecimentos indispensáveis para os SADs atuais. São eles o AFS (Andrew File System) da Universidade de Carnegie-Mellon apoio da IBM, cujo objetivo principal foi projetar um SAD para ambientes acadêmicos. O CODA (Constante Data Availability) também da mesma universidade, possui objetivo focado em alta disponibilidade para aqueles computadores que se desconectavam da rede com mais frequência. Logo após, veio o SPRITE, que se caracterizava mais como um SO distribuído, voltado para operações em rede, desenvolvido pela Universidade da Califórnia, com observação para operações em *cache*, disponibilidade e desempenho [5].

Outros SADs englobam características mais generalistas sendo possível que sejam feitas configurações de acordo com a carga em questão. Nesta categoria de SADs podem ser citados o seguintes: Lustre, PVFS2, OCFS e GlusterFS. O Lustre foi desenvolvido para ser executado em ambientes heterogêneos. O PVFS2 (Parallel Virtual File System) em que duas funções podem ser desempenhadas ao mesmo tempo por um nó do *Cluster*, a de cliente e servidor na prática, sobre redes TCP/IP e *Infiniband*, porém sem um roteamento entre elas. O OCFS, projetado para armazenar bases de dados da Oracle RAC (Real Application Cluster), com o objetivo principal na melhoria de acesso concorrente [2]. E o GlusterFS, SAD mais recente em comparação aos outros citados.

## 2.2. Gluster Filesystem (GlusterFS)

O GlusterFS como descrito em [2], é um SAD que caracteriza-se pela inserção de sistema de arquivos diretamente no *kernel* do SO por meio da API FUSE (*USErspace*), que através de um dispositivo virtual faz a locação entre as chamadas do sistema a esse dispositivo virtual criado, por meio do VFS (*Virtual Filesystem*) do GNU/Linux.

O GlusterFS tem como principal objetivo a escalabilidade, observando também características de disponibilidade e desempenho, podendo ser executado sobre vários tipos de SOs como o Linux, FreeBSD, OpenSolaris e Mac OS X [6]. É importante observar que mesmo com todas essas características do GlusterFS, esta pesquisa limita seu escopo a fazer uma análise quantitativa do desempenho deste SAD. Sua arquitetura é composta pelos componentes básicos, Servidor, Cliente e Volumes, como mostra a Figura 1.

Nesta arquitetura o cliente define o ponto de montagem executando o *Glusterfs-client*, agindo como um centralizador para *bricks* (unidades básicas de armazenamento remotas atribuída ao volume). Servidores (*Nodes*) constituem pontos de armazenamento local aos quais abrigam os locais físicos dos *bricks* para que sejam disponibilizados para os clientes em volumes lógicos. E por fim os volumes, que são caracterizadas por unidades finais às quais são mapeados pelos clientes depois de passarem pela configuração dos tradutores (*Translators*) com as especificações neles definidas [7].

Tradutores são objetos que conectam um ou mais sub-volumes e são carregados em tempo de execução que por meio de uma interface muito bem definida, referenciadas

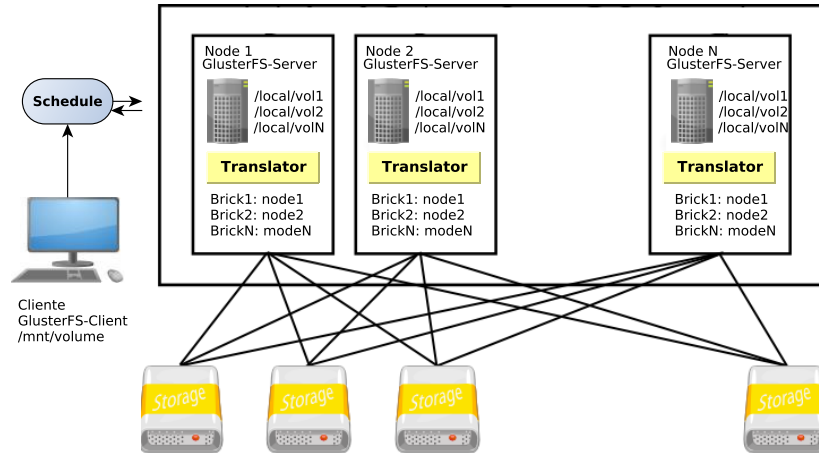


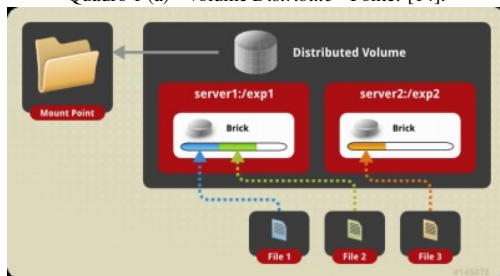
Figura 1. Arquitetura do GlusterFS - Fonte: Elaborada pelos autores

tanto pelos clientes como pelo servidores, caracterizando as funções básicas de cada tipo de volume e suas operações específicas [6]. Outro componente importante do GlusterFS é o *Schedule*, responsável pela definição da maneira como os arquivos serão disponibilizados entre os componentes do sistema e juntamente com os tradutores, possibilita a classificação dos tipos de volumes existentes, que são *Distribute*, *Replicate* e *Stripe*. Estes volumes possuem características peculiares e de acordo com a documentação oficial do GlusterFS, apresentam comportamentos distintos em função dos tradutores.

### 2.3. Volumes avaliados

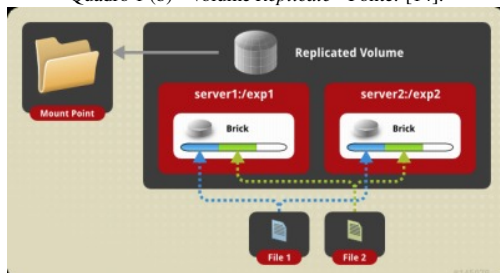
O objeto de avaliação é dividido em três tipos de volumes a serem montados em locais lógicos no cliente GlusterFS, apontando para volumes que mapeiam diretórios físicos nos servidores. Cada volume possui características definidas pelos tradutores do GlusterFS conforme pode ser visto no Quadro 1.

Quadro 1 (a) - Volume *Distribute* - Fonte: [14].

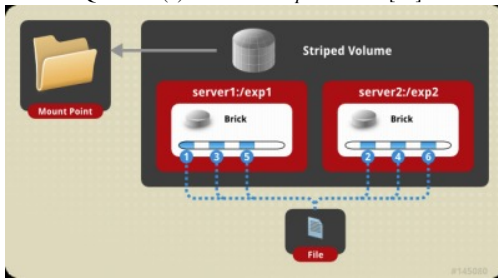


O primeiro volume é o *Distribute* montado no ponto lógico `/mnt/store1` apontando para o volume `node01:vol-store1`. Neste volume as operações de arquivos são feitas de maneira aleatória entre os *bricks*. É considerado pela literatura como um volume para alto desempenho e sem redundância [14].

Quadro 1 (b) - Volume *Replicate* - Fonte: [14].



O segundo tipo de volume montando no local lógico `/mnt/store2` com ponto físico em `node01:/storage/store2` e com o volume denominado de `node01:vol-store2`, foi o *Replicate*. É mais indicado para propósito geral, alta disponibilidade e rapidez em operações de leitura onde as operações de arquivos são feitas de maneira a replicar uma cópia de cada arquivo em todos os *bricks* pertencentes ao volume [14].

Quadro 1 (c) - Volume *Stripe* - Fonte: [14].

O último é o *Stripe*, indicado para ambientes de leitura concorrente, com baixo desempenho para escrita e sem redundância alguma. O volume teve como ponto de montagem o diretório `/mnt/store3`, nome de volume `node01:/vol-store3` e local `/storage/store3`. Para este tipo de volume as operações são realizadas em forma de listras nos *bricks* salvando partes dos arquivos divididos entre os *bricks* do volume [14].

### 3. Trabalhos Relacionados

SADs são fundamentais para diminuição da lacuna existente em E/S de dispositivos de armazenamento e o rendimento de processadores atuais, que a cada dia oferecem mais desempenho [8]. Não obstante, esses sistemas de arquivos vêm sendo tema de diversos trabalhos em importantes congressos em vários países.

No trabalho de Goldman e Carvalho [9], os autores mostram que o SADs PVFS2 foi mais eficiente que o sistema de arquivo local EXT3, mesmo para o caso de apenas uma máquina como cliente. A proposta aqui realizada se diferencia na quantidade de nós envolvidos e no tipo de teste do SAD.

É realizado por Boufleur et al. [6], assim como neste trabalho, a utilização de uma ferramenta de *benchmark* para verificar o desempenho do GlusterFS em relação ao NFS. No trabalho citado, o objetivo principal foi avaliar a utilização do GlusterFS em um *Cluster*, investigando várias situações de cargas geradas pelo *benchmark* BAD e ao final um comparativo com o NFS. As principais diferenças entre os trabalhos são: a ferramenta de *benchmark*, BAD, e nas contribuições de cada trabalho.

Em [2] é realizado um estudo experimental sobre avaliação de SADs em um ambiente de pequena escala, no qual foram avaliados os principais SADs, a fim de identificar seu comportamento no cenário proposto. Esse trabalho diferencia-se do aqui proposto, uma vez que o objetivo principal dele foi mostrar qual o melhor SAD dado seus cenários, ao passo que o trabalho aqui descrito procurou-se comparar os volumes do GlusterFS.

Em [11] é mostrado a importância do estudo e identificação dos comportamentos de padrões de acesso à um SAD. Por meio desta análise, fornece ferramentas para que aplicações possam adequar suas operações de E/S. No trabalho é apresentado um estudo sobre o Lustre por meio de testes que simulam os padrões de acesso. O trabalho, assim como este, se preocupa com o tipo da carga a ser submetida ao *Cluster*. Não obstante, eles se divergem quanto ao objeto a ser testado, no caso, o SAD GlusterFS e na categorização.

Em 2011 e 2012 foram desenvolvidos estudos em SADs por Huaiming et al.[8] e Mendez et al. [12] respectivamente. No primeiro é feita uma proposta de coordenação de E/S do lado do servidor, permitindo um trabalho coordenado por um grupo de servidores atendendo requisições de um aplicativo de forma concorrente, reduzindo assim o tempo de conclusão da tarefa. O trabalho publicado em 2012 [12], mostra a dificuldade da avaliação desse tipo de sistema, em *Cluster*, pela heterogeneidade de sua arquitetura e também complexidade em vários níveis de camadas. Para minimizar este problema foi proposta uma metodologia para avaliar a utilização do esquema E/S dos SADs, levando em consideração o comportamento das aplicações paralelas e a capacidade de E/S.

#### 4. Planejamento da avaliação

Para esta avaliação, foram utilizados os ambientes de teste: *Fast Ethernet* com 100 Mbps e *Gigabit* de 1 Gbps, todos com quatro nodos. Os ambientes foram submetidos a testes por meio da ferramenta de *benchmark IOzone*.

A ordem de execução dos testes foi definida de acordo com o tamanho dos arquivos agrupados em escalas, para assim categorizar as cargas em pequenas, médias e grandes. Cada escala foi submetida aos dois ambientes nos três volumes avaliados.

As escalas foram divididas conforme a Tabela 1, onde são mostradas as operações a que o GlusterFS foi submetido. As colunas representam as operações de *Write* e *Write/Read*, o parâmetro entre parêntese é o responsável por cada tipo de teste do IOzone. O intervalo de crescimento de cada carga pode ser visto na coluna “passo” juntamente com o seu respectivo parâmetro do *benchmark*. No caso o passo abrangeu desde 10 *Kilobytes* até 10 *Gigabytes*, possibilitando desta forma que sejam gerados gráficos com a curva de progressão do comportamento do *benchmark*.

O objetivo desses testes é obter o comportamento das diversas cargas propostas nos três volumes avaliados. Desta maneira, procura-se identificar o comportamento do GlusterFS, para prover melhor desempenho em redes comuns a ambientes de pequenas empresas. Para tal, foram submetidas cargas geradas com tamanhos similares a aplicações encontradas nestes ambientes por meio da ferramenta da *benchmark IOzone*.

Passo (-r)	Write (-i 1)	Write/Read (-i 0 -i 1)
10k	10k a 100k	10k a 100k
100k	100k a 1000k	100k a 1000k
1M	1M a 10M	1M a 10M
10M	10M a 100M	10M a 100M
100M	100M a 1000M	100M a 1000M
1G	1G a 10G	1G a 10G

**Tabela 1. Descrição da carga dos testes em Bytes**

O IOzone, ferramenta de *Benchmark* utilizada nos testes, permite que sejam disparadas várias operações de escrita e leitura com uma série de parâmetros que foram utilizados com cada teste conforme vistos na Tabela 1. A ferramenta permite que sejam criados arquivos em sua execução cujo tamanho total e blocos são configurados em seus parâmetros no momento da chamada do IOzone [13]. Foram criados *scripts* para automatização dos testes, que permitem a determinação da quantidade de execuções do mesmo teste e o cálculo da média, que no caso foram vinte execuções para cada teste.

##### 4.1. Ambientes avaliados

A configuração do GlusterFS foi realizada de maneira a disponibilizar os três volumes testados. Para facilitar a utilização, foram criados diretórios locais em cada servidor e na máquina cliente/servidor no local onde foram executados os testes foi desenvolvida uma configuração que permitiu que sejam montados os três volumes do GlusterFS simultaneamente, o que viabilizou a continuidade dos serviços ao fazer a montagem dos volumes e troca de contexto automaticamente.

Os dois ambientes montados levaram em consideração a mesma quantidade de nós. Para isso foram utilizadas 4 máquinas com processador Intel Core i5 2400, 3.10

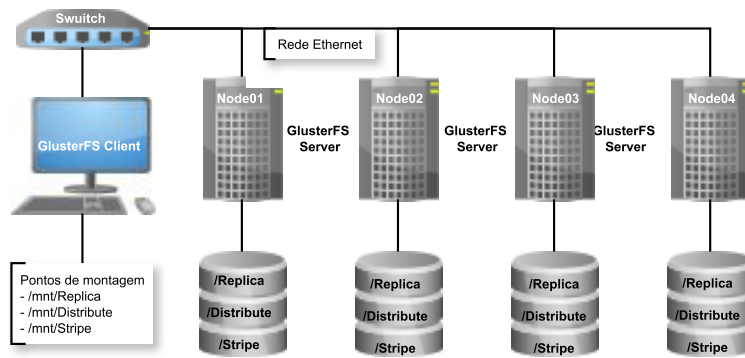


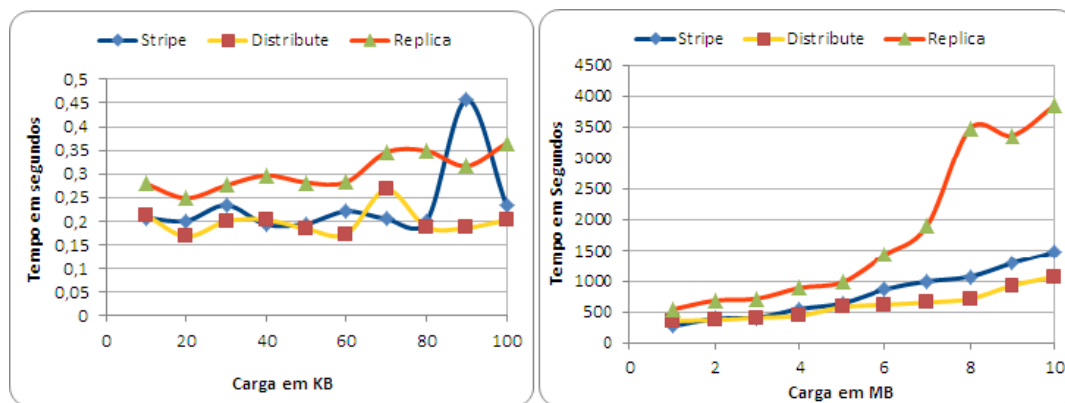
Figura 2. Estrutura montada - Fonte: Elaborada pelos autores

GHz com 4 núcleos reais, com 4 GB de memória RAM, e um disco de 500 GB 7200 RPMS. Para conectividade foi utilizado um *Switch 3Com Gigabit* para o ambiente de 1 Gbps e outro *Switch* de 100 Mbps da mesma marca. A Figura 2 apresenta a estrutura física dos ambientes montados de forma genérica, onde podem ser vistos os componentes: cliente GlusterFS, switch, rede de conectividade, nodos servidores (servidores GlusterFS) e os volumes estudados nos discos de armazenamento.

## 5. Resultados

### 5.1. Resultados quanto ao ambiente de Rede

Para avaliação quanto ao ambiente usou-se um exemplo das classificações de menor e maior tamanho de cada um das tecnologias de rede, comparando seus respectivos resultados a fim de mostrar o comportamento no mesmo cenário para tamanhos de arquivos discrepantes. Os resultados podem ser vistos nas Figuras 3(a) e 3(b).



(a) 10 a 100 kB

(b) 1 a 10 GB

Figura 3. Ambiente *Fast Ethernet* (100 Mbps)

No cenário *Fast Ethernet*, observou-se que o volume *Distribute* apresentou menor tempo de execução para cargas menores. Com destaque para um *outlier* do volume *Stripe* na Figura 3(a), onde a carga de 90 kB foi a pior em todos os casos deste teste.

As Figuras 4(a) e 4(b) referem-se ao cenário com 1 Gbps. Sendo que o volume *Replicate* obteve um pior desempenho para todos os casos, isso porque este volume replica em todos os nós do *Cluster* uma cópia dos arquivos, onerando o tempo de conclusão.

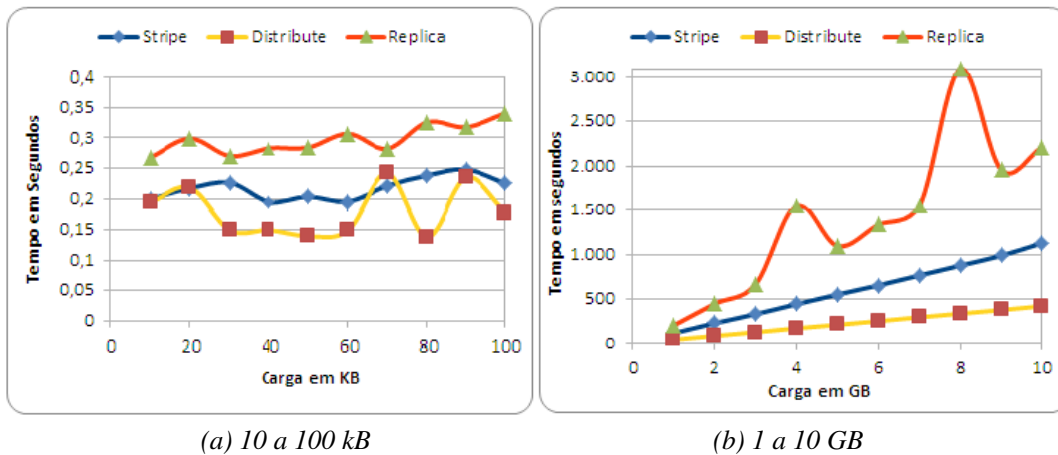


Figura 4. Ambiente *Gigabit* (1 Gbps)

Na ótica dos resultados por tipo de conectividade, percebe-se que o ambiente com 1 Gbps apresenta melhor tempo de conclusão para o *Distribute*. E também uma proximidade de tempo de conclusão para todos os volumes em operações para os arquivos pequenos que é justificada na literatura [2] pela influência do *cache* de cada nó do *Cluster*. Na ocasião, Barbosa e Greve [2] sugeriram que os testes fossem realizados com o tamanho dos arquivos no mínimo igual ao dobro da capacidade a memória RAM. Não obstante, optou-se pela realização desse teste para verificar o comportamento das configurações distintas de cada volume, haja vista que no trabalho citado, testou-se apenas a configuração nativa do GlusterFS que é o volume *Distribute*.

Nos outros casos o ambiente *Gigabit*, mostrou um desempenho bem a cima dos demais, principalmente quando o valor da carga foi aumentando, com destaque para um desempenho de até 50% mais rápido em relação ao ambiente com rede 100 Mbps.

## 5.2. Resultados por tamanho de arquivos

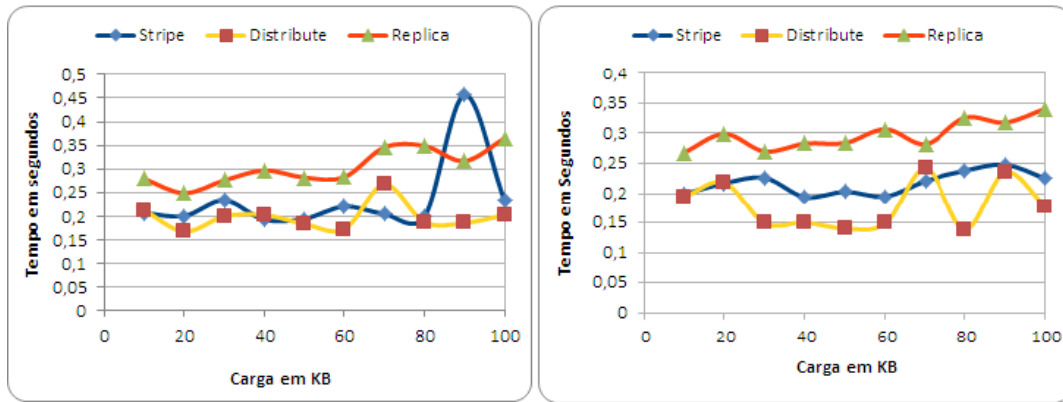
Esta seção apresenta os resultados em relação aos tamanhos de arquivos definidos nos testes. Mostra também a classe de operações empregada no teste que foi o *Escrita/Leitura* (na Tabela 1: *Write/Read*), obtidos pelo uso concomitante dos parâmetros do IOzone,  $-i 0 - i 1$ , que criam um arquivo e executam a sua leitura após a criação.

### 5.2.1. Arquivos pequenos (10 kB até 1000 kB)

Foram realizados testes para arquivos pequenos de 10 a 100kB e 100 a 1000kB. Apresenta-se aqui os testes de 10 a 100 kB conforme Figuras 5(a) e 5(b). Foram desconsiderados testes menores a 10 kB, pois a proximidade nos valores encontrados foi muito pequena para todos os volumes, isso devido a influência da memória *cache* dos computadores envolvidos.

Na Figura 5(a) observa-se pequenas diferenças no tempo de conclusão dos três volumes, onde o *Replicate* ocupou a maior parte do tempo. Já no *Gigabit*, conforme apresentado na Figura 5(b), verifica-se alguns níveis de divergência, onde o melhor caso foi o *Distribute*, e o pior o *Replicate*. Ainda na faixa de testes para *workloads* consideradas pe-

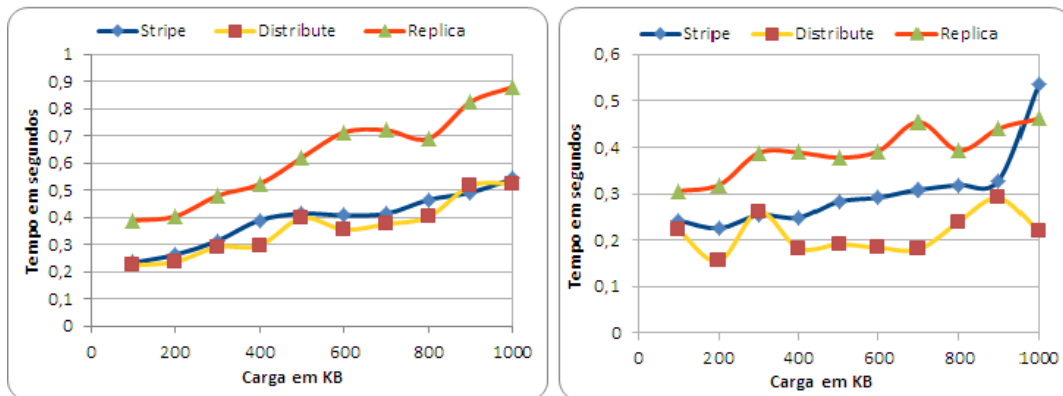




(a) *Fast Ethernet* (b) *Gigabit*  
**Figura 5. Escrita/Leitura de 10 kB até 100 kB**

quenas, verifica-se o desempenho nas Figuras 6(a) e 6(b) o comportamento deste tamanho de arquivos nos dois ambientes propostos.

É possível ver nas Figuras 6(a) e 6(b) que o volume *Replicate* obteve mais uma vez o pior resultado, em consequência da necessidade de replicação de todos os arquivos em todos os *Bricks* do GlusterFS.



(a) *Fast Ethernet* (b) *Gigabit*  
**Figura 6. Escrita/Leitura de 100 kB até 1000 kB**

### 5.2.2. Arquivos médios (1 MB até 100 MB)

As diferenças no teste com arquivos médios no *Fast Ethernet* apresentaram uma inversão, em que o *Replicate* começa com melhor desempenho e termina como o pior. Já no caso do ambiente *Gigabit*, verificou-se que existem diferenças acentuadas, onde o melhor caso para arquivos pequenos foi o volume *Distribute* e o pior caso o volume *Replicate*.

Os testes apresentados nas Figuras 7(a) e 7(b) mostram a submissão de carga com tamanho entre 1 MB e 10 MB nos ambientes. No *Fast Ethernet* apresentado na Figura 7(a) verificou-se um crescimento quase linear até a metade das cargas com todos obtendo desempenho parecido, seguido de uma queda e logo após uma piora do *Replicate*.

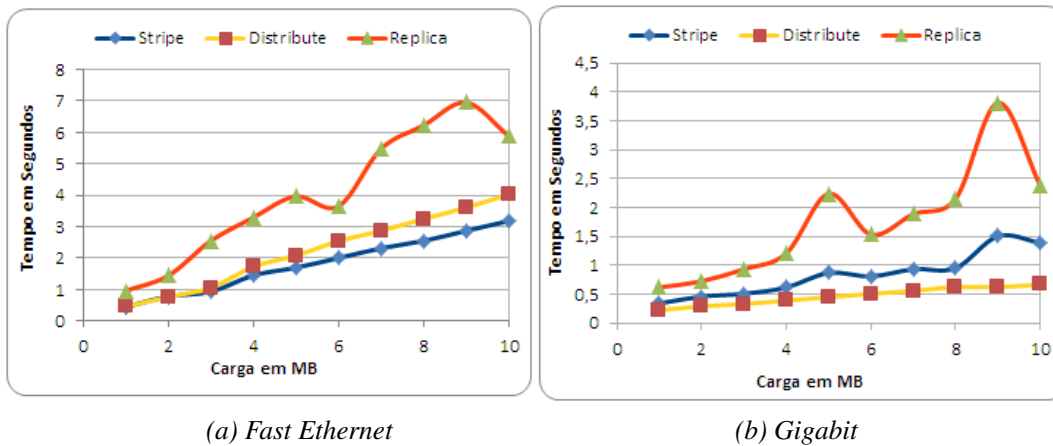


Figura 7. Escrita/Leitura de 1 MB até 10 MB

Observa-se também conforme Figura 7(b) um desempenho similar entre os volumes para a primeira metade dos testes, e para a segunda metade bem superior para os volumes *Distribute* e *Stripe*. No caso, o volume *Replicate* manteve-se parecido com exceção da carga de 9 MB que chegou a um limite de 25 segundos de tempo de execução.

A Figura 8(a) mostra o comportamento das operações de escrita e leitura para as cargas de 10 MB até 100 MB para *Fast Ethernet*, onde todos os volumes foram mais constantes no desempenho em relação ao ambiente *Gigabit* apresentado na Figura 8(b).

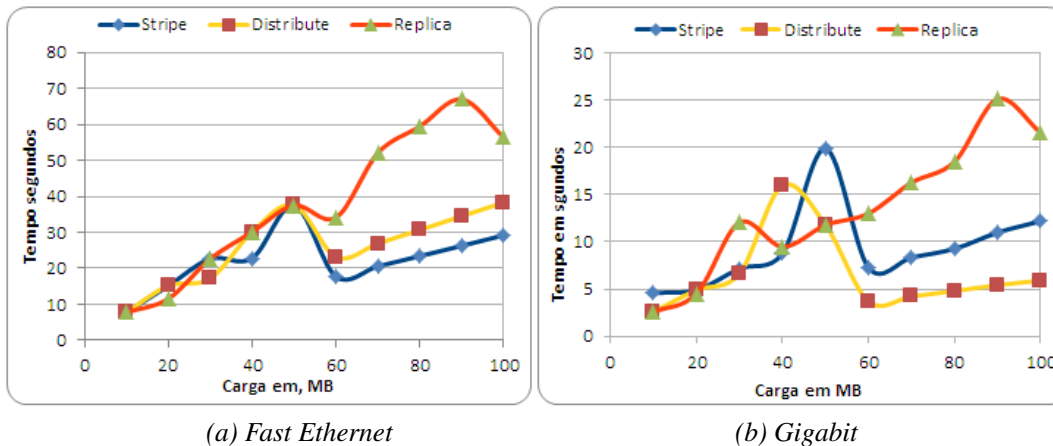


Figura 8. Escrita/Leitura de 10 MB até 100 MB

### 5.2.3. Arquivos grandes (100 MB até 10 GB)

São ilustrados nas Figuras 9(a) e 9(b) os valores para carga de 100 MB a 1000 MB, categorizados como arquivos grandes. Mais uma vez, o ambiente *Fast Ethernet* apresentou diferenças significativas para o volume *Replicate* e certa similaridade para os outros dois volumes, conforme visto na Figura 9(a). Já no ambiente mostrado na Figura 9(b), houve uma grande diferença no tempo de execução do *Replicate* em relação a todos os outros volumes, sendo que o comportamento foi pior neste volume. O melhor volume nesse teste foi o *Distribute*, com quase o dobro do desempenho do mais próximo, o *Stripe*.

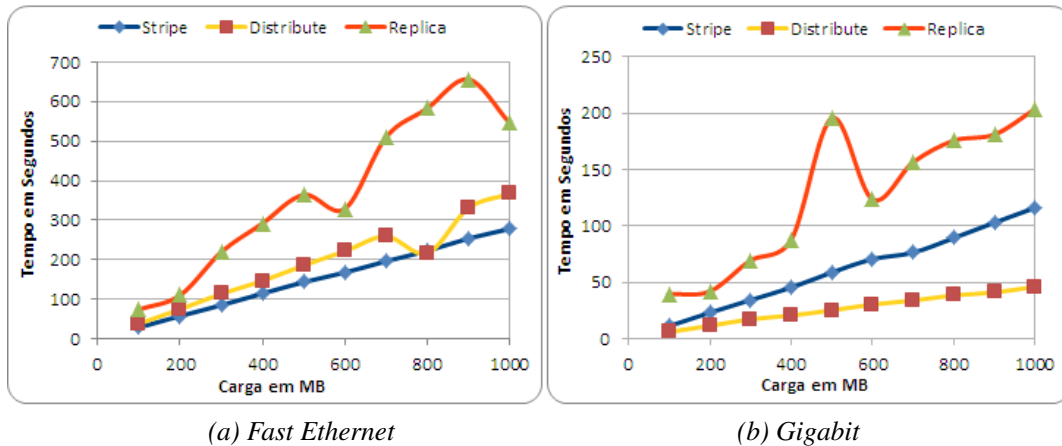


Figura 9. Escrita/Leitura de 100 MB até 1000 MB

São ilustrados nas Figuras 10(a) e 10(b) os tempos de execução para o teste com carga de 1 GB a 10 GB. Neste caso observa-se que para ambos os ambientes o volume *Replicate* apresenta o pior desempenho, sendo que essa diferença é mais tímida no início da carga, mas se tornando bem pior ao final. É possível ver nas Figuras 10(a) e 10(b) que para essa classe de arquivos grandes o rendimento da rede *Gigabit* foi superior.

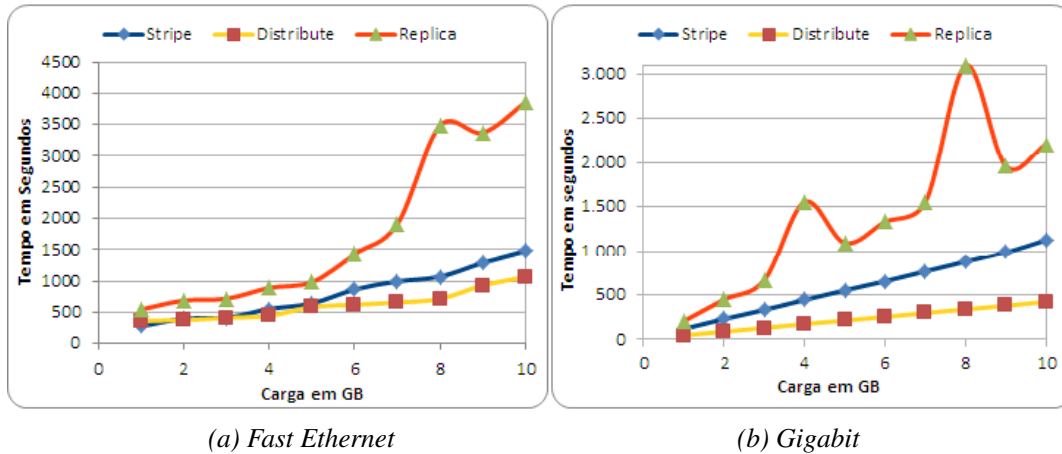


Figura 10. Escrita/Leitura de 1 GB até 10 GB

## 6. Conclusão

Neste trabalho, avaliou-se de forma quantitativa o GlusterFS, levando em consideração aspectos da carga de trabalho nos tipos de volumes utilizados e a rede no qual o SAD foi executado. Para tal avaliação, foram montados dois ambientes, com 100 Mbps e 1 Gbps, cenários mais comuns em empresas de pequeno e médio porte, ambos com a mesma quantidade de nós e submetidos a testes através da ferramenta de *benchmark* IOzone.

Os resultados obtidos demonstraram que o desempenho de cada ambiente é fortemente dependente do tamanho da carga. Pelo tipo de operação realizada, como foi visto na seção de resultados, observa-se comportamentos distintos para os três volumes sendo que, para cargas muito pequenas o volume *Replicate* teve o pior desempenho e o *Distribute* o melhor. Para as maiores, o *Replicate* obteve desempenho significativos por causa da operação de leitura embutida no teste Escrita/Leitura. Quanto às cargas de trabalho,

os resultados apontaram que o volume *Distribute* obteve o melhor desempenho para uma carga média. Não obstante, o volume *Replicate* se mostrou uma boa alternativa para cargas com muitas operações de leitura. Já o *Stripe* no contexto geral foi considerado o pior, como esperado, tendo em vista que ele é recomendando para operações concorrentes.

No contexto geral, o GlusterFS mostrou-se uma ferramenta poderosa para prover desempenho para ambientes em redes avaliados. Entretanto, deve-se ressaltar uma boa análise para identificação das características dos dados que serão utilizados e também as principais características da carga de trabalho, com isso permitir que sejam tomadas decisões em conformidade com o comportamento de cada volume avaliado.

## Referências

- [1] Noronha, R. & Panda, D. K. (2008). IMCa: A High Performance Caching Front-End for GlusterFS on InfiniBand. *Proceedings of the Parallel Processing. ICPP '08. 37th International Conference on*, IEEE, 462-469.
- [2] Barbosa-Jr, A. A.; Greve, F. & Barreto, L. P. (2007). Avaliação de SAD num ambiente de pequena escala. *Anais do XXVII Congresso da SBC*, 852-862.
- [3] Méndez, S.; Rexachs, D. & Luque, E. (2011). Methodology for Performance Evaluation of the Input/Output System on Computer Clusters. *Proceedings of the 2011 IEEE International Conference on Cluster Computing*, IEEE Computer Society, 474-483.
- [4] Wang et al. (2004). File System Workload Analysis For Large Scale Scientific Computing Applications. *21st IEEE/12th NASA Goddard Conference on Mass Storage Systems and Technologies*, College Park, Maryland, USA, 1-14.
- [5] Carvalho R. P. (2005). Sistemas de Arquivos: Paralelos Alternativas para redução do gargalo no acesso ao sistema de arquivos. *Dissertação de Mestrado IME-USP*, 113.
- [6] Bouffleur, M. P. et al. (2007). Avaliação do sistema de arquivos distribuído GlusterFS. *I Concurso de Trabalhos de Iniciação Científica em Arquitetura de Computadores e Computação de Alto Desempenho*, WSCAD-CTIC, 1-4.
- [7] Gluster Community. *GlusterFS Concepts*. <http://gluster.org> acesso: 09/09/2014.
- [8] Huaiming, S.; Yin, Y.; Sun, X.-H.; Thakur, R. & Lang, S. (2011). Server-side E/S coordination for parallel file systems. *Proceedings of 2011 Int. Conference for High Performance Computing, Networking, Storage and Analysis*, ACM, 17:1-17:11.
- [9] Goldman, A. & de Carvalho, R. P. (2007). Sistema de Arquivos Paralelos: Alternativas para a redução do gargalo no acesso ao sistema de arquivos. *WSCAD-CTD*.
- [10] Kunkel, J. M. & Ludwig, T. (2007). Performance Evaluation of the PVFS2 Architecture Parallel. Distributed and Network-Based Processing. *PDP '07. 15th EUROMICRO International Conference on*, 509-516.
- [11] Boito, F. Z.; Kassick, R. V. & Navaux, P. O. A. (2009) Análise de desempenho do sistema de arquivos Lustre sob padrões de acesso a dados exclusivos. *X Simpósio em Sistemas Computacionais*, WSCAD-SSC, 169-176.
- [12] Méndez, S.; Rexachs, D. & Luque, E. *Evaluating Utilization of the I/O System on Computer Clusters*. *WORLDCOMP - The World Congress in Computer Science, Computer Engineering, and Applied Computing*, 12, 2012, 1-7.
- [13] IOzone (2013). File system benchmark. <http://www.iozone.org>, acesso: 09/09/2014.
- [14] Red Hat Storage. *Red Hat Inc*. <http://access.redhat.com>, acesso: 10/09/2014.