

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL  
INSTITUTO DE INFORMÁTICA  
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO

PROJETO PARA O EDITAL MCT/CNPQ NO 27/2007

**Um Modelo de Gerenciamento para  
Grandes Quantidades de Dados em  
Ambientes Largamente Distribuídos**

Profa. Dra. Luciana Porcher Nedel  
Coordenadora

Prof. Dr. Cláudio F. R. Geyer  
Orientador

Diego da Silva Gomes  
Aluno

Porto Alegre, maio de 2008

## SUMÁRIO

<b>1</b>	<b>DADOS DE IDENTIFICAÇÃO</b>	<b>3</b>
1.1	Entidade Proponente	3
1.2	Coordenador	3
1.3	Orientador	3
1.4	Candidato	3
<b>2</b>	<b>CONTEXTO E HISTÓRICO</b>	<b>4</b>
<b>3</b>	<b>MOTIVAÇÃO, ESTADO ATUAL E QUESTÕES DE PESQUISA</b>	<b>5</b>
3.1	Estado da Arte	5
3.2	Questões de Pesquisa	7
<b>4</b>	<b>OBJETIVOS</b>	<b>9</b>
4.1	Objetivo Geral	9
4.2	Objetivos Específicos	9
4.3	Contribuição	9
<b>5</b>	<b>PLANO DE TRABALHO</b>	<b>10</b>
5.1	Metodologia	10
5.2	Cronograma das Atividades	12
<b>6</b>	<b>RESULTADOS ESPERADOS</b>	<b>13</b>
	<b>REFERÊNCIAS</b>	<b>14</b>

# **1 DADOS DE IDENTIFICAÇÃO**

## **1.1 Entidade Proponente**

Programa de Pós-Graduação em Computação – PPGC  
Instituto de Informática  
Universidade Federal do Rio Grande do Sul – UFRGS  
Conceito Capes: 6

## **1.2 Coordenador**

Profa. Dra. Luciana Porcher Nedel  
Universidade Federal do Rio Grande do Sul – Instituto de Informática  
Campus do Vale – Bloco IV – Prédio 43425  
Av. Bento Gonçalves, 9500. Caixa Postal 15064. CEP 91501-970  
Porto Alegre – RS – Brasil  
(51) 3308-7037  
nedel@inf.ufrgs.br

## **1.3 Orientador**

Prof. Dr. Cláudio F. R. Geyer  
Universidade Federal do Rio Grande do Sul – Instituto de Informática  
Campus do Vale – Bloco IV – Prédio 43425  
Av. Bento Gonçalves, 9500. Caixa Postal 15064. CEP 91501-970  
Porto Alegre - RS - Brasil  
(51) 3308-7746  
geyer@inf.ufrgs.br

## **1.4 Candidato**

Diego da Silva Gomes  
Universidade Federal do Rio Grande do Sul – Instituto de Informática  
Campus do Vale – Bloco IV – Prédio 43425 – Sala 207  
Av. Bento Gonçalves, 9500. Caixa Postal 15064. CEP 91501-970  
Porto Alegre – RS – Brasil  
(51) 9312-9920  
dsgomes@inf.ufrgs.br

## 2 CONTEXTO E HISTÓRICO

Atualmente em construção no Centro Europeu para Pesquisas Nucleares (Cern) encontra-se o acelerador de partículas *Large Hadron Collider* (LHC) (LHC: THE LARGE HADRON COLLIDER, 2008). Quando entrar em operação, os experimentos que dele fazem parte começarão a coletar massivas quantidades de dados com o objetivo de fazer a síntese de novas informações. Por exemplo, procurar-se-á nesses dados evidências sobre a existência da partícula *Higgs Boson*. Caso verificada, comprovariam-se grandes teorias sobre a origem da matéria e do universo (BUNN; NEWMAN, 2003). Somente em um dos detectores de partículas, o *Compact Muon Solenoid* (CMS) (NEWBOLD, 1994), serão produzidos cerca de 5 *Petabytes* de dados anualmente. Estima-se que, juntamente com o experimento ATLAS, tenha-se ultrapassado 1 *Exabyte* ( $1 \text{ Exabyte} = 10^{18} \text{ Bytes}$ ) em dados produzidos já na próxima década (BUNN; NEWMAN, 2003).

Para poder armazenar, processar e analisar esse grande volume de dados, adotou-se como plataforma de execução as **Grades Computacionais** (FOSTER; KESSELMAN, 1999; FOSTER; KESSELMAN; TUECKE, 2001; FOSTER, 2002; FOSTER; TUECKE, 2005). O termo **Grade** advém de uma metáfora onde os recursos computacionais estariam tão facilmente acessíveis quanto aqueles da grade da rede elétrica. No caso dos experimentos da Física de Altas Energias, as Grades são usadas para distribuir cópias (réplicas) dos dados coletados (atualmente são resultados de simulações), formando-se as chamadas **Grades de Dados** (CHERVENAK et al., 2000; HOSCHEK et al., 2000).

### 3 MOTIVAÇÃO, ESTADO ATUAL E QUESTÕES DE PESQUISA

Nas Grades de Dados, utilizam-se ferramentas para catalogar e transferir as réplicas dos arquivos, tais como aquelas que acompanham o *toolkit* do Globus (FOSTER; KESSELMAN, 1997): GridFTP (ALLCOCK et al., 2005), RLS (CHERVENAK et al., 2002) e o RFT (ALLCOCK; FOSTER; MADDURI, 2004). Essas ferramentas constituem os chamados **Sistemas de Gerenciamento de Réplicas** (*Replica Management Systems – RMS*) (VENUGOPAL; BUYYA; RAMAMOHANARAO, 2006).

O uso dos RMS requer que as réplicas sejam armazenadas em servidores dedicados, de grande capacidade e controlados por presença humana (CAMARGO; KON, 2007). Essas exigências surgem da incapacidade dos RMS de lidar automaticamente com as falhas dos servidores, de dar garantias quanto à disponibilidade e consistência das réplicas e dos metadados utilizados para sua localização; e de lidar com aspectos relativos à segurança. Como os arquivos são replicados na íntegra, faz-se necessário uma grande quantidade de espaço de armazenamento, além de exigir redes de alta velocidade para dar vazão ao grande volume de dados produzidos (BUNN; NEWMAN, 2003). Como essa solução impõe um alto custo, dificulta-se ainda mais o envolvimento de países em desenvolvimento como o Brasil nesses projetos, já que a construção dos experimentos, por si só, também apresenta um custo muito elevado.

Por outro lado, existe no ambiente de Grade uma grande quantidade de recursos menos capacitados e com menor disponibilidade, mas que estão freqüentemente ociosos. Por exemplo, pode-se citar os *desktops* que os cientistas utilizam para acessar a Grade, estações de trabalho e pequenos *clusters* localizados em laboratórios nas instituições onde esses cientistas se encontram (DABEK et al., 2001; ROWSTRON; DRUSCHEL, 2001a; ADYA et al., 2002). De fato, são nesses recursos que os dados são processados após os arquivos de entrada terem sido obtidos dos servidores dedicados que rodam os RMS.

O problema consiste então em agregar um potencial grande número de recursos menos capacitados, mais instáveis, não confiáveis, não dedicados e distribuídos globalmente em uma solução para o gerenciamento das grandes quantidades de dados.

#### 3.1 Estado da Arte

Devido à grande escala e à necessidade de tolerar um grande número de falhas, as soluções de gerenciamento de dados em ambientes largamente distribuídos têm caminhado no sentido da descentralização de tarefas, focando em aspectos de auto-gerenciamento e auto-manutenção. Como o modelo **par-a-par** (P2P) (ANDROUTSELLIS-THEOTOKIS; SPINELLIS, 2004; COULOURIS; DOLLIMORE; KINDBERG, 2005) lida diretamente

com esses aspectos, ele tem sido muito utilizado nas soluções mais atuais, aliado a técnicas para melhorar a distribuição e a disponibilidade dos dados. Exemplos dessas técnicas incluem o uso de nodos virtuais (STOICA et al., 2001), de algoritmos de dispersão da informação (IDA) (RABIN, 1989), de fragmentação e de *caching*.

A maneira como um modelo P2P e essas técnicas são empregadas depende de objetivos para o gerenciamento dos dados, do ambiente computacional (recursos) disponível, das exigências feitas para os arquivos e de características dos usuários. Na Grade de Dados da Física de Altas Energias, por exemplo, existem poucas fontes produzindo arquivos, que na maioria das vezes são imutáveis ou com poucas exigências de escrita (BUNN; NEWMAN, 2003). Nesse caso, não há necessidade de fornecer mecanismos para controle de concorrência de escrita. Há, porém, a necessidade por controlar o acesso aos dados e fazer a auditoria do espaço utilizado por cada usuário. Precisa-se também que os arquivos sejam duráveis, apesar do grande número de falhas aos quais as máquinas estão susceptíveis. Já num sistema para compartilhamento de arquivos, tais como o Gnutella (GNUTELLA, 2008), não existe preocupação nenhuma quanto à persistência dos dados, podendo acontecer de alguns arquivos se tornarem indisponíveis por longos períodos ou mesmo deixarem de existir. Nesses tipos de sistema de gerenciamento de dados P2P, freqüentemente os usuários são anônimos e não há controle do espaço utilizado.

Devido aos requisitos específicos de cada caso, existe uma grande quantidade de trabalhos envolvendo o tema da gerência de dados em ambientes largamente distribuídos. No contexto de Grade, no entanto, as possibilidades são menores, pois abordagens baseadas nos RMS ainda são muito utilizadas. Nesse sentido, escolheu-se para análise os trabalhos que, de alguma forma, abordem aspectos de gerência de dados em Grade e que utilizem um modelo P2P.

No OppStore (CAMARGO; KON, 2007, 2006), as máquinas são organizadas em *clusters* que formam uma rede P2P estruturada baseada no algoritmo Pastry (ROWSTRON; DRUSCHEL, 2001b). O modelo utiliza replicação para os metadados principais, algoritmos de dispersão da informação (IDA) para os arquivos e lida com a heterogeneidade de recursos através do uso de **identificadores virtuais** (CAMARGO; KON, 2007). Existe nele uma forte preocupação em garantir uma alta disponibilidade para os dados, ainda que aspectos de segurança sejam negligenciados.

CFS (DABEK et al., 2001), por outro lado, propõe uma solução que relembra sistemas de arquivos distribuídos. Ela consiste em fragmentar os arquivos em blocos, distribuindo-os entre máquinas organizadas em uma rede P2P pelo algoritmo Chord (STOICA et al., 2001). Os blocos podem ainda ser interpretados como metadados, referindo uma estrutura de diretório como nos sistemas de arquivos convencionais. Apesar de distribuir bem os dados, o fato de utilizar um tamanho de bloco único e pequeno impõe uma baixa disponibilidade para os dados.

Já no PAST (ROWSTRON; DRUSCHEL, 2001a), os arquivos são armazenados e replicados na íntegra, tal como ocorre com o uso dos RMS. No entanto, a durabilidade dos dados é garantida de forma automatizada. As máquinas, apesar de estarem em um ambiente institucional, formam uma rede P2P baseada no algoritmo Pastry. A replicação também é utilizada por propósitos de otimização. A grande vantagem do PAST em relação aos demais é a sua forte preocupação com relação à segurança. Sua desvantagem diz respeito à ausência de fragmentação e a necessidade por *smartcards*, características que impõem um maior custo à solução.

Contudo, a impraticabilidade dessas soluções no contexto da Grade da Física se dá basicamente devido ao grande volume de dados envolvido. Nela, os arquivos atingem

com facilidade a ordem de *Gigabytes* em tamanho, exigindo mecanismos que minimizem a quantidade de operações de manutenção que movimentam dados de uma máquina para outra. Precisa-se também que as movimentações, quando necessárias, sejam projetadas para aumentar a vazão dos dados, diminuindo-se assim o tempo necessário para sua conclusão.

Outra característica peculiar à Grade da Física diz respeito à grande heterogeneidade dos recursos de armazenamento disponíveis no ambiente que se forma. Torna-se necessário agregar máquinas com pouca capacidade, ainda que utilizando de forma eficiente os recursos de armazenamento em massa, tais como aqueles baseados em discos e fitas. Precisa-se ainda atender às exigências por segurança e por disponibilidade e durabilidade para os dados.

Para ser usado no ambiente de Grade, torna-se necessário que a solução seja **interoperável** com outros serviços fornecidos pela plataforma. Exemplos desses serviços incluem escalonadores, ferramentas para transferência de arquivos e serviços de monitoramento. Devido ao grande volume de dados, precisa-se fornecer mecanismos de busca e ferramentas para o uso dos serviços que sejam voltadas ao usuário final.

Em trabalho recentemente realizado no Grupo de Processamento Paralelo e Distribuído (GPPD) da Universidade Federal do Rio Grande do Sul (UFRGS), propôs-se um sistema para o gerenciamento de dados da Grade da Física de Altas Energias, chamado JavaRMS (PETEK et al., 2006). O trabalho foi desenvolvido como parte do doutorado de Marko Petek (ainda em andamento) em conjunto com o trabalho de mestrado de Diego Gomes (já concluído) (GOMES, 2008). JavaRMS é o resultado de uma cooperação existente entre o GPPD e o subgrupo de Caltech que desenvolve *software* de Grade para o LHC. Essa cooperação se dá através do físico Alberto Santoro da Universidade Estadual do Rio de Janeiro (UERJ).

JavaRMS foi especialmente desenvolvido para grandes volumes de dados, ainda que atendendo às exigências feitas pela Grade da Física. A arquitetura prevê desde a comunicação segura e interoperável com os outros serviços da Grade até os sistemas de busca voltados para o usuário final. O sistema permite agregar recursos de baixa capacidade com facilidade, de forma a reduzir os custos para a construção da Grade. Seu projeto para arquivos grandes permitiu melhorar significativamente o desempenho de aplicações intensivas em dados em diversos contextos. Os resultados mostraram ainda que o modelo é escalável e atende às necessidades por segurança e durabilidade dos dados.

Apesar do JavaRMS ter satisfeito às exigências básicas para o gerenciamento dos dados da Grade da Física, ainda há necessidade por muita pesquisa para que, de fato, a solução atenda plenamente todos os requisitos.

### 3.2 Questões de Pesquisa

Durante o desenvolvimento do JavaRMS, identificou-se diversas questões de pesquisa passíveis de serem tratadas em um trabalho de doutorado:

- *Tolerância a falhas*: os serviços da arquitetura carecem de pesquisa para atender a aspectos de tolerância a falhas, tais como consistência de réplicas e resiliência, confiabilidade e robustez dos serviços oferecidos;
- *Popularidade dos dados*: um dos aspectos não abordados no JavaRMS diz respeito à popularidade dos dados. Arquivos com maior demanda exigem um maior número

de servidores envolvidos no seu gerenciamento a fim de distribuir a carga de atendimento às requisições. Por outro lado, a demanda é variável com o tempo e precisa de uma solução adaptativa para lidar de forma eficiente com a popularidade dos dados;

- *Disponibilidade dos dados*: para dar garantias quanto à disponibilidade para os dados gerenciados, precisa-se pesquisar o comportamento dos nodos que entram e saem da rede (a disponibilidade das máquinas). Estudos preliminares indicam que, no ambiente-alvo do sistema, a predição de tal característica é um problema difícil. Trabalhos relacionados apontam para pesquisas envolvendo também soluções adaptativas;
- *Proximidade de rede na distribuição dos dados*: um dos aspectos determinantes para o desempenho das funções que operam sobre a grande quantidade de dados diz respeito à proximidade destes em relação à máquina que os necessita. Apesar de ser possível lidar com a proximidade da rede física nos algoritmos P2P, ela só pode ser considerada durante a etapa de localização dos dados. Torna-se necessário, portanto, pesquisar algoritmos alternativos que formam a rede lógica P2P que considerem a proximidade de rede durante a fase de distribuição dos dados. Um exemplo de estrutura de dados que poderia ser fornecido numa solução nesse sentido é o de *skiplists* (HARVEY et al., 2003);
- *Serviços de busca*: embora a arquitetura tenha previsto mecanismos de busca por metadados, a serem desenvolvidos como parte do trabalho do doutorando Marko Petek, estes serviços carecem de maior abstração para serem utilizados junto a ferramentas voltadas ao usuário final.



## 4 OBJETIVOS

### 4.1 Objetivo Geral

O objetivo deste projeto é a amadurecer a solução JavaRMS, tratando de aspectos não antes desenvolvidos no contexto do projeto devido à complexidade a eles inerente, a fim de que atenda na plenitude às exigências de gerenciamento de dados da Grade da Física de Altas Energias. Pretende-se, assim, reduzir os custos para a construção das Grades, permitindo maior participação do Brasil em experimentos como o acelerador de partículas LHC.

### 4.2 Objetivos Específicos

Como objetivos específicos, destaca-se:

- Fazer um levantamento das técnicas de tolerância a falhas empregadas em sistemas de gerência de dados para Grades;
- Tratar do problema imposto pela popularidade dos dados através de uma solução adaptativa;
- Dar garantias quanto à disponibilidade dos dados gerenciados;
- Fornecer uma mídia de armazenamento global para aplicações de Grade colaborativas;
- Fazer a conexão entre os serviços de busca e as ferramentas voltadas ao usuário final;
- Agregar as extensões à solução JavaRMS já em funcionamento;
- Analisar o possível ganho no desempenho ao lidar com a proximidade de rede durante a fase de distribuição dos dados, comparando-a com a solução anterior.

### 4.3 Contribuição

A principal contribuição deste trabalho é um modelo de sistema de gerência de dados para ambientes largamente distribuídos capaz de lidar com grandes volumes de dados, ainda que adaptando-se às variações de demanda pelos arquivos e tolerando falhas no fornecimento de seus serviços.

## 5 PLANO DE TRABALHO

### 5.1 Metodologia

Para atingir os objetivos, pretende-se realizar as seguintes etapas de trabalho:

1. **Requisitos para o doutorado:** objetiva atender às exigências impostas pelo programa de doutorado do PPGC/UFRGS;
  - (a) Revalidação dos 24 créditos cursados no mestrado;
  - (b) Complementação dos créditos para o doutorado;
  - (c) Realização da atividade didática durante o estágio-docência;
  - (d) Apresentação do andamento da pesquisa na Semana Acadêmica/PPGC;
  - (e) Prestação do exame de qualificação em abrangência;
  - (f) Apresentação e defesa da proposta de tese;
2. **Pesquisa Bibliográfica:** objetiva identificar o estado da arte.
  - (a) Análise de artigos que propõem técnicas de tolerância a falhas para a gerência de dados em ambientes largamente distribuídos;
  - (b) Análise de trabalhos envolvendo a predição de disponibilidade das máquinas;
  - (c) Análise de técnicas usadas para melhorar a disponibilidade dos dados;
  - (d) Estudo e comparação das técnicas de replicação automática e *caching* para determinar qual é a mais adequada para o tratamento da popularidade dos dados;
  - (e) Estudo sobre *skiplists* para a distribuição dos dados;
3. **Definição do Modelo:** objetiva modelar os componentes da arquitetura responsáveis pelos novos serviços, e remodelar os componentes já existentes reconsiderando aspectos de tolerância a falhas.
  - (a) Remodelagem dos componentes, aplicando-se as técnicas de tolerância a falhas estudadas na etapa anterior;
  - (b) Definição dos novos componentes necessários para atender às novas exigências;
  - (c) Remodelagem das operações envolvendo arquivos sobre a nova estrutura de dados distribuída (*skiplists*);

4. **Prototipação:** objetiva implementar as funcionalidades sobre a arquitetura do JavaRMS.

- (a) Implementação de um componente para controle de popularidade dos dados;
- (b) Acoplamento de serviços de Grade para a monitoria da disponibilidade das máquinas;
- (c) Implementação de um componente para controle da disponibilidade dos dados;
- (d) Recodificação de outros componentes considerando aspectos de tolerância a falhas;
- (e) Substituição do componente que lida com a formação da rede lógica P2P por outro que considere a localidade de rede durante a distribuição dos dados;
- (f) Cooperação internacional com outro grupo de pesquisa (sanduíche) a fim de trocar experiências e identificar outras necessidades para amadurecimento da solução. Possibilidades: Cern (Genebra) ou Grenoble.

5. **Validação:** objetiva validar as funcionalidades desenvolvidas.

- (a) Instalação do novo sistema sobre a plataforma fornecida pela grade francesa Grid5000 (GRID'5000, 2008);
- (b) Análise de artigos envolvendo técnicas de teste para validar aspectos de tolerância a falhas e disponibilidade dos dados;
- (c) Implementação dos testes de tolerância falhas e disponibilidade dos dados;
- (d) Reexecução de testes de desempenho com a nova estrutura de distribuição dos dados;
- (e) Análise do espaço de armazenamento ocupado e da capacidade de atender a requisições frente às diferenças de popularidade dos arquivos;

6. **Redação:** objetiva planejar e redigir o texto da tese, bem como divulgar o trabalho desenvolvido por meio de artigos.

- (a) Redação da tese;
- (b) Redação de artigos para publicação nacional (WSCAD) e internacional (SBAC, CCGrid, IPDPS) e periódicos internacionais (*Journal of Grid Computing*);

## 5.2 Cronograma das Atividades

Atividade	2008/2	2009/1	2009/2	2010/1	2010/2	2011/1	2011/2	2012/1
1a	X							
1b	X	X						
1c	X							
1d			X		X		X	
1e			X					
1f				X				
2a	X							
2b	X							
2c	X	X						
2d		X						
2e		X						
3a		X						
3b			X					
3c				X				
4a				X				
4b				X				
4c					X			
4d					X	X		
4e						X		
4f							X	
5a					X			
5b					X			
5c						X		
5d						X		
5e							X	
6a							X	X
6b			X		X		X	X

## 6 RESULTADOS ESPERADOS

Do ponto de vista científico, espera-se:

- Poder agregar com maior facilidade as máquinas ao sistema, diminuindo-se assim o custo da solução. Essa característica será o resultado do uso de serviços tolerantes a falhas;
- Obter um ganho no desempenho das operações que movimentam dados através da utilização de uma estrutura distribuída capaz de considerar a localidade de rede durante a distribuição dos dados;
- Poder atender a uma maior demanda pelos dados. Como essa demanda seria atendida adaptativamente através do controle de popularidade dos dados, não seria necessário manter um grande número de cópias dos arquivos, diminuindo-se também os custos com armazenamento;
- Melhorar a disponibilidade dos dados e reduzir os custos de armazenamento através do uso de algoritmos de dispersão da informação (IDA);
- Fornecer uma maior autonomia de uso através do acoplamento dos serviços de busca junto às ferramentas de usuário. Assim, o sistema poderia ser utilizado de forma mais parecida como aquela idealizada pela metáfora de Grade, onde os recursos computacionais estariam tão acessíveis como aqueles da rede elétrica.

Em relação à formação do doutor envolvido com a efetivação deste trabalho, o resultado esperado é de que ele desenvolva estratégias para lidar com problemas complexos. Por envolver um grande número de variáveis, espera-se que o doutor em questão se torne capaz de identificar e selecionar para o estudo os fatores de maior relevância.

## REFERÊNCIAS

ADYA, A. et al. FARSITE: Federated, Available, and Reliable Storage for an Incompletely Trusted Environment. In: SYMPOSIUM ON OPERATING SYSTEMS DESIGN AND IMPLEMENTATION, 5., 2002, Boston, Massachusetts, USA. **Proceedings...** New York: ACM Press, 2002. p.1–14.

ALLCOCK, W. et al. The Globus Striped GridFTP Framework and Server. In: CONFERENCE ON SUPERCOMPUTING, 2005. **Proceedings...** Washington: IEEE Computer Society, 2005. p.54.

ALLCOCK, W.; FOSTER, I.; MADDURI, R. Reliable Data Transport: A Critical Service for the Grid. In: GLOBAL GRID FORUM, BUILDING SERVICE BASED GRIDS WORKSHOP, 11., 2004, Honolulu, Hawaii, USA. **Proceedings...** Open Grid Forum, 2004.

ANDROUTSELLIS-THEOTOKIS, S.; SPINELLIS, D. A Survey of Peer-to-Peer Content Distribution Technologies. **ACM Computing Surveys**, New York, NY, USA, v.36, n.4, p.335–371, Dec. 2004.

BUNN, J. J.; NEWMAN, H. B. Data Intensive Grids for High Energy Physics. In: GRID COMPUTING: MAKING THE GLOBAL INFRASTRUCTURE A REALITY, 2003. **Proceedings...** [S.l.]: John Wiley & Sons, 2003. (Wiley Series in Communications Networking & Distributed Systems).

CAMARGO, R. Y. de; KON, F. Distributed Data Storage for Opportunistic Grids. In: INTERNATIONAL MIDDLEWARE DOCTORAL SYMPOSIUM, MDS, 3., 2006, Melbourne, Australia. **Proceedings...** New York: ACM Press, 2006.

CAMARGO, R. Y. de; KON, F. Design and Implementation of a Middleware for Data Storage in Opportunistic Grids. In: INTERNATIONAL SYMPOSIUM ON CLUSTER COMPUTING AND THE GRID, CCGRID, 7., 2007, Rio de Janeiro, Brazil. **Proceedings...** [S.l.]: IEEE Computer Society, 2007. p.23–30.

CHERVENAK, A. et al. The Data Grid: Towards an Architecture for the Distributed Management and Analysis of Large Scientific Datasets. **Journal of Network and Computer Applications**, Oxford, United Kingdom, v.23, n.3, p.187–200, July 2000.

CHERVENAK, A. et al. Giggle: A Framework for Constructing Scalable Replica Location Services. In: CONFERENCE ON SUPERCOMPUTING, 2002, Baltimore, Maryland, EUA. **Proceedings...** [S.l.]: IEEE Computer Society, 2002. p.1–17.

COULOURIS, G.; DOLLIMORE, J.; KINDBERG, T. **Distributed Systems - Concepts and Design**. 4.ed. Reading, USA: Addison-Wesley, 2005.

DABEK, F. et al. Wide-area cooperative storage with CFS. In: SYMPOSIUM ON OPERATING SYSTEMS PRINCIPLES, SOSP, 18., 2001, Banff, Alberta, Canada. **Proceedings...** New York: ACM Press, 2001. p.202–215.

FOSTER, I. What Is the Grid? A Three Point Checklist. **Grid Today**, San Diego, CA, USA, v.1, n.6, p.22–25, July 2002.

FOSTER, I.; KESSELMAN, C. Globus: A Metacomputing Infrastructure Toolkit. **International Journal of Supercomputer Applications and High Performance Computing**, Cambridge, MA, USA, v.11, n.2, p.115–128, Summer 1997.

FOSTER, I.; KESSELMAN, C. Computational Grids. In: FOSTER, I.; KESSELMAN, C. (Ed.). **The Grid**: blueprint for a new computing infrastructure. San Francisco, CA, USA: Morgan Kaufmann, 1999. p.15–51.

FOSTER, I.; KESSELMAN, C.; TUECKE, S. The Anatomy of the Grid: Enabling Scalable Virtual Organizations. **International Journal of High Performance Computing Applications**, Thousand Oaks, CA, USA, v.15, n.3, p.200–222, 2001.

FOSTER, I.; TUECKE, S. Describing the Elephant: The Different Faces of IT as Service. **ACM Queue**, New York, NY, USA, v.3, n.6, p.26–29, July/Aug. 2005.

GNUTELLA. Disponível em: <<http://www.gnutella.com>>. Acesso em: fevereiro 2008.

GOMES, D. S. **JavaRMS**: um sistema de gerência de dados para grades baseado num modelo par-a-par. 2008. Dissertação (Mestrado em Ciência da Computação) — Instituto de Informática, Universidade Federal do Rio Grande do Sul, Porto Alegre.

GRID'5000. Disponível em: <<http://www.grid5000.fr>>. Acesso em: fevereiro 2008.

HARVEY, N. J. et al. SkipNet: A Scalable Overlay Network with Practical Locality Properties. In: USENIX SYMPOSIUM ON INTERNET TECHNOLOGIES AND SYSTEMS, USITS, 4., 2003, Seattle, WA, USA. **Proceedings...** [S.l.]: USENIX Association, 2003. v.4.

HOSCHEK, W. et al. Data Management in an International Data Grid Project. In: INTERNATIONAL WORKSHOP ON GRID COMPUTING, GRID, 1., 2000, Bangalore, India. **Proceedings...** London: Springer-Verlag, 2000. p.77–90. (Lecture Notes in Computer Science, v.1971).

LHC: The Large Hadron Collider. Disponível em: <<http://lhc.web.cern.ch/lhc/>>. Acesso em: fevereiro 2008.

NEWBOLD, D. **The Compact Muon Solenoid – Technical Proposal**. Switzerland: CERN, European Laboratory for Particle Physics, 1994. (CERN/LHCC 94-38 and CERN/LHCC-P1).

PETEK, M.; GOMES, D. S.; STEENBERG, C.; GEYER, C. F. R.; DIVERIO, T. A.; SANTORO, A. A Model to Implement a Files and Replicas System in Clarend. In: INTERNATIONAL WORKSHOP ON HIGH-PERFORMANCE DATA MANAGEMENT

IN GRID ENVIRONMENTS, HPDGRID, 2006, Rio de Janeiro, RJ, Brazil. **Proceedings...** Springer-Verlag, 2006.

RABIN, M. O. Efficient Dispersal of Information for Security, Load Balancing, and Fault-Tolerance. **Journal of the ACM (JACM)**, New York, NY, USA, v.36, n.2, p.335–348, April 1989.

ROWSTRON, A.; DRUSCHEL, P. Storage Management and Caching in PAST, a Large-Scale, Persistent Peer-to-Peer Storage Utility. **Operating Systems Review**, New York, NY, USA, v.35, n.5, p.188–201, Dec. 2001. Trabalho apresentado no ACM Symposium on Operating Systems Principles, 18., 2001, Alberta, Canada.

ROWSTRON, A.; DRUSCHEL, P. Pastry: Scalable, decentralized object location and routing for large-scale peer-to-peer systems. In: IFIP/ACM INTERNATIONAL CONFERENCE ON DISTRIBUTED SYSTEMS PLATFORMS, MIDDLEWARE, 18., 2001, Heidelberg, Germany. **Proceedings...** Berlin: Springer-Verlag, 2001. p.329–350. (Lecture Notes in Computer Science, v.2218).

STOICA, I. et al. Chord: A Scalable Peer-to-peer Lookup Protocol for Internet Applications. In: SPECIAL INTEREST GROUP ON DATA COMMUNICATIONS, SIGCOMM, 2001, San Diego, CA, USA. **Proceedings...** [S.l.: s.n.], 2001. p.149–160.

VENUGOPAL, S.; BUYYA, R.; RAMAMOHANARAO, K. A Taxonomy of Data Grids for Distributed Data Sharing, Management, and Processing. **ACM Computing Surveys**, New York, NY, USA, v.38, n.1, p.1–53, March 2006.