

# Compartilhamento de Dados e Recursos Computacionais de Armazenamento em Redes P2P Sociais

Fabício Barros Gonçalves, Carlo Oliveira, Vanessa Braganholo

Programa de Pós-Graduação em Informática (PPGI)  
Universidade Federal do Rio de Janeiro (UFRJ)

fabriciobarros@saocamilo-es.br, carlo@nce.ufrj.br, braganholo@dcc.ufrj.br

**Abstract.** *Management and transparent access to data and computational storage resources in e-Science highly distributed environments is a hard problem. In order to contribute to the solution of this problem, this paper proposes an architectural model to sharing storage resources and collaborative data hosting in social P2P networks.*

**Resumo.** *O gerenciamento e o acesso transparente a dados e aos recursos computacionais de armazenamento em ambientes altamente distribuídos de e-science, é um problema difícil. Para contribuir para a solução deste problema, este trabalho propõe um modelo arquitetural para o compartilhamento de recursos computacionais de armazenamento e hospedagem colaborativa de dados compartilhados em redes P2P sociais.*

## 1. Introdução

O uso de ferramentas avançadas para computação de alto desempenho (por exemplo, grades computacionais), e de sistemas de gerenciamento de workflows têm revolucionado as atividades de pesquisa em diversos domínios como química, física, biologia, medicina, e até mesmo ciências sociais. Nesse cenário, cientistas definem suas hipóteses e criam experimentos com base em um conjunto de algoritmos e programas que operam em ambientes de computação distribuída, permitindo, então, a geração, coleta e armazenamento de um grande montante de dados experimentais. Porém, organizar esses dados em um repositório ao qual pesquisadores possam ter acesso a qualquer hora e de qualquer local requer atividades especializadas, tais como as exercidas por administradores de infraestruturas computacionais, ou a utilização de ferramentas computacionais para lidar com grandes volumes de dados. Nesse sentido, tais ferramentas devem permitir a distribuição de dados sobre recursos computacionais de armazenamento em um ambiente altamente distribuído.

Segundo Mattoso *et al.* [14], o gerenciamento e a transparência de acesso aos recursos computacionais de armazenamento em um ambiente altamente distribuído, com uma boa relação custo/desempenho, é um problema difícil. Além disso, é possível observar a falta de técnicas de computação distribuída que escalem para configurações muito grandes, e, ao mesmo tempo, tratem da autonomia, dinâmica e heterogeneidade dos recursos (ex., processadores, memória, discos, conteúdo, software, etc.) [14]. Autonomia, dinâmica e heterogeneidade remetem a ambientes P2P (*peer-to-peer*) [15]. De fato, este ambiente oferece diversos benefícios, dos quais podemos citar: compartilhamento do custo operacional da infraestrutura de computação; aumento da oferta de recursos em sistemas de computação distribuída; agregação e interoperabilidade de recur-

sos entre sistemas heterogêneos; tolerância a falhas; e colaboração na execução de tarefas complexas, como a manipulação de grandes montantes de dados entre as entidades componentes do sistema distribuído.

Este trabalho foca no problema de compartilhamento de grandes volumes de dados entre pesquisadores que atuem em um mesmo projeto ou área afim. Para isso, propomos utilizar uma rede P2P como infraestrutura de rede e, sobre ela, construir redes sociais [4,10,11] para facilitar a conexão (de forma dinâmica) [17] entre os pesquisadores com interesses em comum. Este trabalho contribui nessa direção propondo uma arquitetura de software para a construção e gerenciamento de tais redes, além do gerenciamento de todo o compartilhamento e armazenamento de grandes volumes de dados pelos pesquisadores que participam dessas redes.

Deste modo, nossa proposta é que redes sociais atuem como nuvens computacionais [5], onde cada *peer* compartilha parte de seus recursos computacionais e hospeda de forma colaborativa os dados compartilhados entre os *peers* com os mesmos interesses. Para que isso aconteça, é necessário responder alguns questionamentos: (i) Como usuários com interesses comuns podem compartilhar recursos e dados em um ambiente distribuído descentralizado de larga escala? (ii) Como criar redes sociais a partir dos relacionamentos sociais dos usuários? (iii) Como os computadores dos usuários, participantes de uma partição de rede social, podem ser utilizados para armazenar colaborativamente os dados compartilhados? (iv) Como dados, mantidos sobre as várias partições de redes sociais no ambiente distribuído, podem ser recuperados de acordo com a necessidade do usuário? A arquitetura proposta neste trabalho é formada por cinco camadas distintas, que possuem componentes necessários para construção de um *middleware* para endereçar os questionamentos levantados acima.

O restante deste artigo está dividido em 4 seções. A Seção 2 apresenta os trabalhos relacionados. As Seções 3 e 4 apresentam um cenário de uso da abordagem proposta e o modelo arquitetural. Por fim, a Seção 5 apresenta as considerações finais.

## 2. Trabalhos Relacionados

Esta seção discute os problemas associados ao compartilhamento de recursos computacionais de armazenamento e hospedagem colaborativa de dados em redes P2P sociais. Em [8] apresentamos uma descrição mais detalhada dos trabalhos relacionados.

**Redes P2P Sociais.** Redes P2P são redes de computadores onde todos os computadores têm as mesmas funcionalidades e papéis. Devido a essas características, soluções algorítmicas foram propostas a fim de construir infraestruturas de redes P2P. Segundo Semenov [16], redes P2P são divididas em duas categorias: redes P2P estruturadas (ex. Chord, CAN, Pastry e Tapestry) e não estruturadas (ex. Gnutella e FreeNet). Apesar de existirem diversas soluções para estruturação de redes P2P, elas não levam em consideração as preferências de usuário. Com isso, algoritmos para construção de redes P2P sociais têm sido propostos a fim de criar redes sociais sobre infraestruturas de redes P2P já conhecidas [4,10,11]. Tais trabalhos possuem uma característica em comum: eles propõem algoritmos que combinam os interesses dos *peers* (especificados via palavras-chave) a fim de selecionar aqueles com maior grau de similaridade.

**Backup Colaborativo de Dados.** A combinação de falhas freqüentes em redes P2P e o longo tempo demandado para replicação de dados, perdidos durante as saídas dos *peers* de uma rede, podem tornar o armazenamento de grandes volumes de dados dispendioso

ou até mesmo impossível em termos de desempenho. Num cenário de redes P2P sociais para armazenamento colaborativo, duas técnicas de redundância de dados, tipicamente utilizadas em sistemas de armazenamento, podem ser utilizadas: replicação e *erasure coding* [9]. Em replicação, cópias exatas dos dados são distribuídas e armazenadas em peers distintos. Já em *erasure coding*, os dados são divididos em blocos e codificados, para que, em seguida, sejam distribuídos e armazenados em peers distintos. A partir de um conjunto de blocos codificados é possível reconstruir os dados originais.

**Controle de Réplicas.** Para manter a disponibilidade de dados em uma rede P2P social, é necessário que réplicas [1,2] dos dados compartilhados sejam mantidas localmente nos computadores dos usuários. Em ambientes distribuídos onde usuários colaboram uns com os outros, compartilhando, acessando e atualizando dados, é necessário o uso de uma abordagem de replicação múltiplos mestres para assegurar a disponibilidade dos dados [13]. Dessa forma, em cenários de *e-science* (alta colaboração), este trabalho adota replicação múltiplos mestres para definir onde os dados serão atualizados. Para definir quando os dados devem ser atualizados, este trabalho adota replicação otimista, pelo fato dessa abordagem permitir a atualização assíncrona das réplicas.

### 3. Cenário de Uso

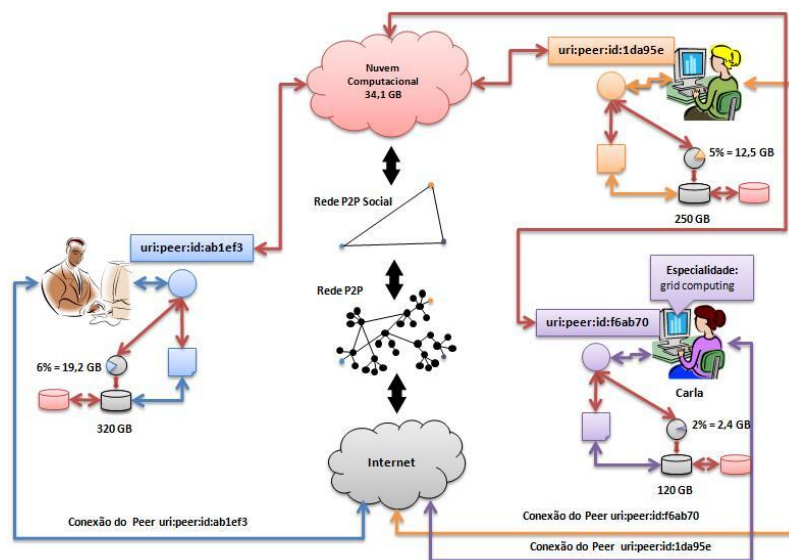
Segundo Mattoso et al. [14], faltam técnicas de computação distribuída que escalem para configurações muito grandes, e, ao mesmo tempo, tratem da autonomia, dinâmica e heterogeneidade dos recursos em ambientes computacionais *e-science*. Com base nisso, um exemplo de cenário de uso da abordagem proposta nesse trabalho é apresentado na Figura 1. Nesse cenário, três pesquisadores (Carla, Marcos e Rita) necessitam compartilhar informações para realização de um experimento científico, que será realizado no centro de pesquisas onde Rita trabalha. Rita precisa compartilhar dados com os demais pesquisadores e vice-versa.

Conforme a Figura 1, os computadores pessoais de cada pesquisador possuem conexão com a Internet. Assim, a proposta desse trabalho é que cada computador pessoal se transforme em um *peer*, isto é, cada computador pode atuar como cliente ou servidor ao mesmo tempo, iniciar uma conexão com outros computadores conectados a uma rede P2P a qualquer instante de tempo e, ainda, compartilhar os recursos computacionais e/ou conteúdos existentes em computadores pessoais. Isso deve permitir que recursos computacionais de armazenamento sejam agregados, para formar nuvem computacional [5], a fim de que os dados compartilhados pelos pesquisadores sejam mantidos nessa entidade computacional.

Quando os computadores pessoais dos pesquisadores se conectam a uma rede P2P, eles recebem um identificador único, construído com base no modelo URN (*Uniform Resource Name*). Com isso, cada computador pessoal passa a fazer parte de uma infraestrutura descentralizada de sistema distribuído. Nela, cada pesquisador pode compartilhar livremente os conteúdos e recursos computacionais de armazenamento de seus computadores. Apesar de nesse cenário somente existirem três pesquisadores, uma rede P2P pode ser formada pelos computadores de outros usuários (note que a rede é representada como um grafo na Figura 1).

Uma vez que seus computadores fazem parte da rede P2P, os pesquisadores podem compartilhar conteúdos e recursos computacionais de armazenamento. Assim, quando um pesquisador compartilha um conteúdo sobre a rede P2P, informações sobre o com-

partilhamento são disseminadas na rede. Nesse momento, as mensagens de publicação de conteúdo procuram *peers* cujo usuário tenha algum interesse sobre o seu conteúdo. Assim, da mesma forma que um usuário compartilha um conteúdo por meio de palavras-chave que classificam o item compartilhado, ele também pode explicitar o seu interesse por algum conteúdo, bastando informar as possíveis palavras-chave relacionadas ao conteúdo idealizado. Com isso, à medida que mensagens com informações de compartilhamento passam pelos *peers*, elas são recebidas e, em seguida, analisadas para verificar se as informações de conteúdo compartilhado satisfazem aos interesses do usuário representado pelo *peer*. Outro ponto importante sobre as mensagens de publicação de conteúdo é que elas, além de carregar informações dos conteúdos compartilhados pelos usuários, carregam informações sobre os interesses dos usuários que compartilham conteúdos. Essas informações são importantes, pois elas ajudam o mecanismo de descoberta e roteamento a tomarem decisões sobre qual o melhor caminho que deve ser seguido por uma mensagem na infraestrutura de rede P2P.



**Figura 1. Cenário de uso**

Após cada pesquisador ter compartilhado conteúdos, especificado seus interesses e recebido as informações de compartilhamento de conteúdo uns dos outros, eles decidem formar uma rede social, como é exibido no cenário apresentado pela Figura 1. As redes sociais, formadas pelos usuários conectados ao sistema distribuído, são criadas a partir da adição de um usuário na lista de relacionamentos de outro usuário, formando pequenos grafos sobre a infraestrutura de rede P2P. É o usuário quem decide se deseja manter algum relacionamento com os demais usuários do sistema. É importante lembrar que essa decisão é tomada com base em alguma afinidade que um usuário tenha com outro. No cenário apresentado nesta seção, têm-se três pesquisadores que fazem parte de um projeto de bioinformática e que desejam acompanhar e estudar os dados de um experimento científico que será realizado por um deles. Quando os relacionamentos são criados, os pesquisadores disponibilizam os conteúdos e, também, um percentual de recurso computacional de armazenamento que desejam compartilhar.

Após os relacionamentos e compartilhamentos terem sido efetivados pelos pesquisadores, um pequeno aglomerado de recursos computacionais de armazenamento é formado, totalizando um espaço compartilhado de 34,1 GB. Esse aglomerado de recursos

denota uma nuvem computacional, que neste trabalho é uma entidade de alto nível, atuando como um agrupamento de recursos computacionais oriundos dos relacionamentos sociais entre usuários do ambiente. Na visão dos pesquisadores, as nuvens são vistas de maneira transparente, ou seja, eles não precisam especificar qualquer tipo de endereço para acessar uma nuvem, pois ela é uma visão nomeada dos relacionamentos entre um pesquisador e os outros e, vice-versa. As visões da nuvem auxiliam o usuário a visualizar seus relacionamentos sociais, além de possibilitar uma visão única do espaço de armazenamento criado de maneira colaborativa. Assim, cada pesquisador pode ter acesso aos conteúdos compartilhados na nuvem. Além disso, quando um conteúdo é compartilhado com um ou mais usuários, esses se tornam também proprietários do conteúdo e passam a hospedar o mesmo. Aqui podem surgir questões de privacidade, que serão tratadas em trabalhos futuros.

Por fim, à medida que novos usuários ingressam em nuvens computacionais, por meio dos relacionamentos sociais com os usuários dessas entidades computacionais, a disponibilidade dos conteúdos compartilhados mais importantes aumenta, pois, como mencionado anteriormente, cada usuário compartilha parte dos seus recursos computacionais para auxiliar a hospedagem de conteúdos. A escala da nuvem cresce conforme o número de relacionamentos sociais entre os usuários da nuvem, ou seja, à medida que os relacionamentos são criados, quantidades de recursos computacionais de armazenamento são ofertadas para uso na nuvem. Outra forma de aumentar a escala da nuvem é por meio do aumento do tamanho do compartilhamento dos recursos computacionais de armazenamento.

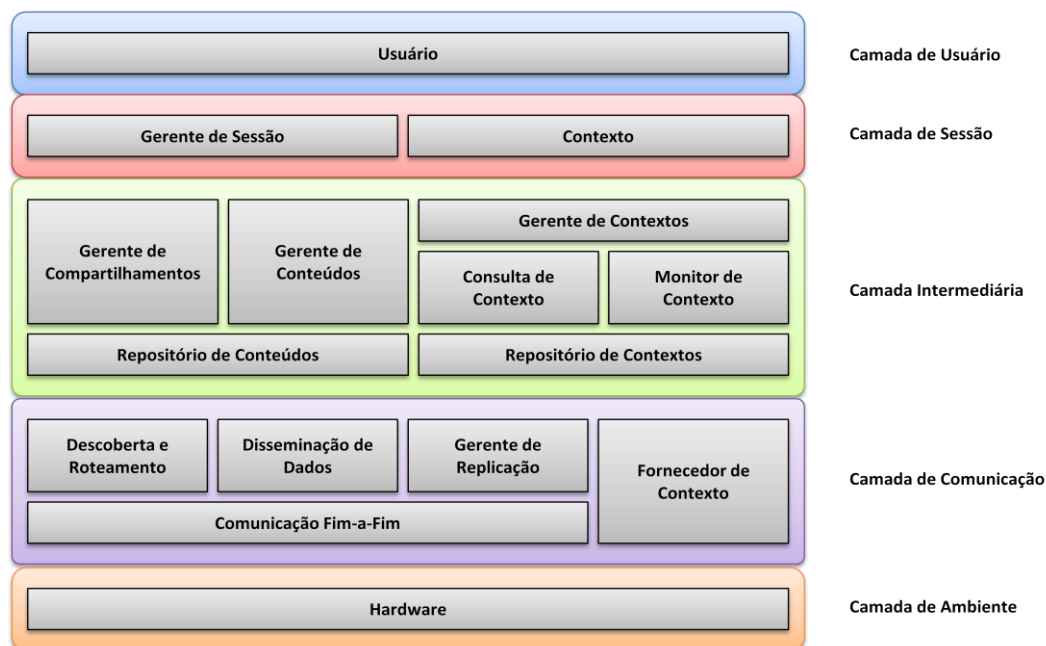
#### 4. Arquitetura Proposta

Atualmente, muito se tem estudado sobre algoritmos para controle de topológico, descoberta de recursos e roteamento de mensagens em redes P2P sociais. No entanto, pouca atenção tem sido dada ao problema de compartilhamento, agregação e utilização de recursos computacionais em redes P2P sociais. Não basta somente criar um conjunto de políticas de sistemas distribuídos a fim de propiciar o compartilhamento de recursos e dados sobre partições de redes P2P (redes P2P sociais). Deste modo, é necessário que arquiteturas de software sejam definidas, visando à definição de componentes básicos, que, por sua vez, ofereçam estruturas de dados para armazenamento de informações oriundas das execuções destas políticas. Além disso, faz-se necessário definir componentes para atuarem como *locus* de implementação das políticas.

Esta seção propõe um modelo arquitetural com objetivo de contribuir para criação de soluções para manipulação de grandes volumes de dados em ambientes altamente distribuídos. Para especificação do modelo, utilizamos os requisitos levantados pelo cenário de uso apresentado pela seção 3. Além disso, foram investigadas algumas soluções arquiteturais [1,6,10] que pudessem guiar a criação das camadas e componentes apresentados pela Figura 2. Mais detalhes sobre a arquitetura são apresentados em [8].

**Camada de Ambiente.** Esta camada contém uma coleção de componentes, representados pelo componente Hardware, que devem fornecer interfaces para controlar recursos computacionais, tais como: discos, adaptadores de redes, etc. Ela serve para esconder detalhes de gerenciamento de arquivos, gerenciamento de memória, etc., e é utilizada para fornecer informações de hardware, como por exemplo, espaço livre em disco, capacidade de disco, etc.

**Camada de Comunicação.** A camada de comunicação é composta por mecanismos para comunicação fim-a-fim, descoberta de nós, roteamento de mensagens, disseminação de informação, gerenciamento de transferência de dados e fornecimento de contexto. Os mecanismos de descoberta e roteamento são representados pelo componente de Descoberta e Roteamento. Assim, quando *peers*, recursos e dados são descobertos na rede P2P, o componente de Descoberta e Roteamento armazena referências para eles, permitindo que uma topologia de rede seja formada à medida que os *peers* se conectam uns aos outros. Dessa forma, uma rede de cobertura é criada e, quando um *peer* envia mensagens para outro *peer*, essas são roteadas até que elas cheguem aos seus destinos.



**Figura 2. Modelo arquitetural.**

O mecanismo para disseminação de informações sobre *peers* e compartilhamentos, tanto de recursos computacionais como de dados, é representado pelo componente de Disseminação de Dados. Para isso, técnicas de disseminação de informação, como *Push*, *Pull* e *Push and Pull*, podem ser utilizadas para publicar informações de *peers* e compartilhamentos na rede P2P. O componente de Disseminação de Dados faz uso das referências para os *peers* e para recursos compartilhados, que são mantidos pelo componente de Descoberta e Roteamento. O componente Gerente de Replicação é responsável pelo envio e recepção de réplicas de dados compartilhados entre usuários. Além disso, ele também controla a atualização de réplicas de dados. Já o componente Fornecedor de Contexto é responsável pela captura de informações de contextos e por direcioná-las para a camada intermediária. Entende-se como contexto qualquer informação que possa ser usada para caracterizar a situação de uma entidade, que pode ser uma pessoa, lugar ou objeto que é considerado importante para a interação entre os usuários do sistema [6].

**Camada Intermediária.** A camada intermediária responsabiliza-se pela gerência e monitoramento de recursos computacionais, conteúdos e contextos. A camada é composta por vários outros componentes. O Gerente de Contexto é um componente para recuperação e atualização de contextos mantidos pelo Repositório de Contextos. A atualização de contexto pode ser feita pelo Monitor de Contexto e Consulta de Contexto. O Monitor

de Contexto atualiza o Repositório de Contexto com informações locais de contexto como, por exemplo, quantidade de unidades de discos, quantidade de espaço livre em unidades de discos, relacionamentos de usuários. Já a Consulta de Contexto atualiza o Repositório de Contexto com informações coletadas pelas consultas remotas (por exemplo, conteúdos que combinam com os interesses do usuário), que são feitas sobre a rede P2P. O gerenciamento dos conteúdos compartilhados é feito utilizando um conjunto de componentes, que é composto pelo Gerente de Compartilhamentos, Gerente de Conteúdos e Repositório de Conteúdos. A função do Gerente de Compartilhamento é manter o compartilhamento dos conteúdos. Quando um compartilhamento é criado, o conteúdo é mantido pelo Repositório de Conteúdos. As funcionalidades para acesso e manipulação de conteúdos compartilhados são disponibilizadas pelo Gerente de Conteúdos. Esse componente acessa e manipula os conteúdos mantidos no Repositório de Conteúdos.

**Camada de Sessão.** Esta camada manipula informações sobre as interações entre o usuário e o ambiente distribuído. A camada de sessão envia e recupera tais informações para a camada intermediária, que por sua vez, as armazenam na forma de contextos.

**Camada de Usuário.** A camada de usuário manipula a identificação e perfil do usuário. Ela é responsável por capturar contextos do usuário como, por exemplo, informações pessoais e interesses pessoais sobre tipos de conteúdos. Esses dados são importantes para interação do usuário com o ambiente distribuído. Assim, interesses do usuário por determinados tipos de conteúdos são utilizados para buscar conteúdos compartilhados e informações sobre usuários do ambiente distribuído. Por fim, informações pessoais são utilizadas para identificar o usuário e, por isso, elas podem ser utilizadas para autenticação de usuários.

## 5. Conclusão

Este trabalho apresentou um modelo arquitetural para compartilhamento de dados e recursos computacionais de armazenamento em redes P2P sociais cujo objetivo é colaborar para a construção de soluções para problemas relacionados a desempenho em *e-science*, atuando especificamente no armazenamento distribuído de grandes volumes de dados. Os componentes do modelo proposto serão utilizados no projeto e construção de um *middleware* que trate os detalhes de políticas para criar nuvens computacionais por meio de relacionamentos sociais entre usuários, gerenciar o compartilhamentos de recursos computacionais, gerenciar o compartilhamento de conteúdos para usuários participantes de uma mesma nuvem computacional e controlar a hospedagem colaborativa de conteúdos sobre os recursos computacionais de armazenamento, a partir da replicação de conteúdos e o controle de réplicas dos mesmos.

Atualmente, as políticas apresentadas acima vêm sendo definidas, prototipadas e simuladas por meio de um simulador (<http://code.google.com/p/pysocialsim/>) desenvolvido pelos autores deste trabalho. O simulador será utilizado para desenvolver, aperfeiçoar e simular as políticas para compartilhamento de dados e recursos computacionais em redes P2P sociais. Juntamente com isso, o simulador também auxiliará no desenvolvimento de provas de conceito do modelo arquitetural proposto neste trabalho. Assim, pretende-se, com o resultado das simulações, obter resultados que possam fornecer dados de escalabilidade e desempenho da solução proposta. Por fim, a partir da prova de conceito do modelo arquitetural, pretende-se desenvolver um *middleware* para oferecer APIs para que ferramentas de manipulação de dados possam ser construídas, sem que o

desenvolvedor tenha que se preocupar com as políticas do ambiente distribuído e de como os conteúdos compartilhados serão hospedados nos recursos computacionais de armazenamento.

## Referências

- [1] Akbarinia, R., Martins, V., Pacitti, E., Valdiriez, P. (2006). Design and Implementation of APPA. Global Data Management, IOS Press.
- [2] Akbarinia, R., Pacitti, E., Valdiriez, P. (2007). Data Currency in Replicated DHTs. In: SIGMOD, pp. 211-222.
- [3] Andrade, N.; Santos-Neto, E.; Brasileiro, F. (2008). Scalable resource annotation in peer-to-peer grids. In: P2P. Aachen, Germany, pp. 231-234.
- [4] Carchiolo, V.; Malgeri, M; Mangioni, G.; Nocosia, V. (2008). Emerging structures of the P2P networks induced by social relationships, Computer Communications, v. 31, pp. 620-628.
- [5] Chappel, D. (2009). A Short Introduction to Cloud Platforms: An enterprise-oriented view. Disponível em: <http://www.davidchappell.com/CloudPlatforms--Chappell.pdf>.
- [6] El-Khatib, K.; Zhang, Z. E.; Hadibi, N.; e van Bochman, G. (2004). Personal and Mobility in Ubiquitous Computing Environment. Wireless Communications and Mobile Computing, v. 4, pp 595-607.
- [7] Figueiredo, R. J.; Boykin, P. O.; Juste, P. S.; Wolinsky, D. (2008). Facilitating the deployment of ad-hoc virtual organizations with integrated social and overlay networks. In: HPDC, Boston, USA, pp 201-204.
- [8] Gonçalves, F.; Oliveira, C.; Braganholo, V. Uma arquitetura para compartilhamento de dados e recursos computacionais de armazenamento em redes P2P Sociais. Relatório Técnico 03/09. Disponível em <http://www.dcc.ufrj.br/~braganholo/artigos/RT-03-09.pdf>
- [9] Li, J.; Dabek, F. (2006). F2F: reliable storage in open networks. In: IPTPS, NY.
- [10] Lin, C. J.; Chang, Y. T.; Tsai, S. C.; Chou, C. F. D. (2007). Distributed social-based overlay adaptation for unstructured P2P networks. In: IEEE Global Internet Symposium, pp 1-6.
- [11] Liu, L.; Antonopoulos, N.; Mackin, S. (2007). Social Peer-to-Peer for resource discovery. In: EUROMICRO, Napoli, Italy, pp 459-466
- [12] Martins, V; Akbarinia, R; Pacitti, E; Valduriez, P. (2006). Reconciliation in the APPA P2P system. In: ICPADS.
- [13] Martins, V.; Pacitti, E.; Valduriez, P. Survey of data replication in P2P systems. Relatório Técnico 6083 , INRIA, França.
- [14] Mattoso, M.; Werner, C.; Travassos, G.; Braganholo, V.; Murta, L. (2008). Gerenciando experimentos científicos em larga escala. In: SEMISH, pp. 121-135
- [15] Milojicic, D. S.; Kalogeraki, V.; Lukose, R.; Nagaraja, K.; Pruyne, J.; Richard, B.; Rollins, S; e Xu, Z. (2002). Peer-to-Peer Computing. HP, Tech. Report HPL-2002-57.
- [16] Semenov, A. (2005). Evolution of Peer-to-peer algorithms: Past, present and future. Seminar on Internetworking, HUT T-110.551, Helsinki University of Technology, 2005.
- [17] Zhou, D.; Lo, V. (2004). Clustering on the fly: resource discovery in a cycle sharing peer-to-peer systems. In: IEEE CCGRID. Chicago, USA, pp 66-73.