Gerenciando Experimentos Científicos em Larga Escala

Marta Mattoso[1], Cláudia Werner[1], Guilherme Horta Travassos[1], Vanessa Braganholo[2], Leonardo Murta[1]

[1]Programa de Engenharia de Sistemas e Computação - COPPE/UFRJ

[2]Departamento de Ciência da Computação – IM/UFRJ

{marta,werner,ght,murta}@cos.ufrj.br, braganholo@dcc.ufrj.br

Abstract. *Several scientific areas, such as bioinformatics and oil engineering, need means of executing simulation-based experiments. The state of the practice for this, in most of the cases, consists in the execution of a set of programs. This, however, is not enough to deal with the complexity imposed by the problems that need to be analyzed. This issue gets worse with large-scale experiments. In this case, we need a system to manage the composition of processes and data in a coherent flux. Also, this system must be capable of registering the steps and parameters used in the well-succeeded executions of the experiment. The main motivation of this paper is in identifying and analyzing the challenges that need to be addressed to provide computational support to the development of large-scale scientific experiments. The challenges we identify here deal with the general problem of managing scientific experiments to several applications and resources distributed over a large-scale network such as grids. We identify three complementary research directions: the performance, the management process, and the semantic support. For each of them, we point out some possible solution paths.*

Resumo. *Diversas áreas científicas, tais como bioinformática e engenharia de petróleo, necessitam de meios para a execução de experimentos baseados em simulação. O estado da prática para esse fim consiste, na grande maioria das vezes, na execução de um conjunto de programas, o que não é suficiente para tratar a complexidade imposta pelos problemas a serem analisados. O problema se agrava quando o experimento ocorre em larga escala. Faz-se necessário um sistema que gerencie a composição de processos e dados num fluxo coerente, e que registre as etapas realizadas com escolhas de parâmetros de execuções bem sucedidas do experimento. A motivação principal desse artigo está em identificar e analisar os desafios necessários para prover apoio computacional ao desenvolvimento de experimentos em larga escala. Os desafios que destacamos aqui lidam com o problema geral de gerência de experimentos científicos para várias aplicações e com recursos distribuídos em uma rede de larga escala como grids. Identificamos nesse problema três vertentes de pesquisa complementares na gerência do experimento científico, a saber: o desempenho, o processo de gerência e o apoio semântico, para os quais apontamos algumas direções de possíveis soluções.*

1. Introdução

Os procedimentos computacionais têm sido grandes aliados de físicos, químicos, geólogos e cientistas de um modo geral. Entretanto, à medida que o conhecimento avança, um grande volume de dados se apresenta disponível para complementar as análises desses processos. A informação científica continua a ser gerada em experimentos ditos físicos (bancada de laboratórios) e, mais recentemente, a partir de redes de sensores, mas também se apresenta proveniente de análises obtidas com simulações computacionais, visualização e com tarefas de mineração de dados. Esse conjunto de dados gera um volume enorme de informação fundamental na análise de processos científicos.

Fazer ciência hoje implica, dentre outros aspectos, ubiqüidade e distribuição, visando ao desenvolvimento e execução de soluções com alto desempenho, baseadas em reutilização, gerência de dados e experimentos, por exemplo. Daí surge o termo e-ciência (ou *e-science*), que caracteriza o apoio ao cientista para o desenvolvimento de ciência em larga escala utilizando infra-estrutura computacional correspondente. Conseqüentemente, sem essa infra-estrutura, o pesquisador que faz ciência corre o risco de não avançar na sua pesquisa na velocidade desejada, perdendo competitividade frente ao progresso da ciência no cenário mundial.

Processos experimentais isolados, interligados apenas na concepção do cientista que conduz a análise, não são atualmente suficientes para tratar a complexidade imposta pelos problemas a serem analisados. O problema se agrava quando o experimento científico ocorre de modo distribuído e em larga escala. Faz-se necessário um sistema que gerencie a composição de processos e dados num fluxo coerente, descrito através de um *workflow* científico, e que registre as etapas realizadas com escolhas de parâmetros de execuções bem sucedidas do experimento, independente do local de execução.

Um experimento científico no contexto deste trabalho engloba todo o processo de modelagem, execução e análises de processos científicos de forma encadeada e controlada. A Figura 1 mostra etapas do ciclo de vida de um experimento científico, de acordo com a visão do projeto myGrid/myExperiment (Goble e De Roure, 2007). Um experimento científico com tal apoio visa a ser reproduzível, permitindo o rastreamento (proveniência) dos processos que levaram a um determinado resultado, facilitando assim o compartilhamento e compreensão do processo científico por diversos pesquisadores.

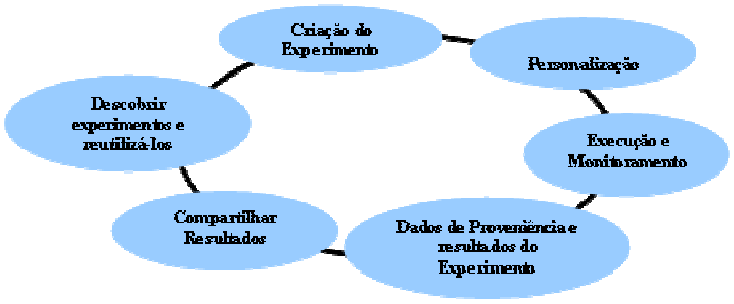

Figura 1. Ciclo de vida de um experimento do myGrid (Goble e De Roure, 2007)

A criação do experimento através da concepção dos *workflows* científicos necessita de processos de apoio, que permitam o reaproveitamento de partes de

workflows anteriores e conhecimentos relacionados na construção de novos *workflows*. As áreas de reutilização de software e gerência de conhecimento têm muito a contribuir para que esse reaproveitamento de *workflows* científicos anteriores ocorra de forma sistemática e controlada. Esta idéia vai ao encontro de Shawn Bowers que diz que: *workflows* científicos precisam, em primeiro lugar, "ajudar os cientistas a realizarem seu trabalho, para que deste modo eles possam focar em fazer ciência" (Bowers, 2007).

Diversas iniciativas foram criadas na direção do apoio à ciência em larga escala. Em especial, podem-se destacar os programas governamentais de apoio a e-ciência nos EUA (*cyberinfraestructure*) e Reino Unido (*e-science*) (UK e-Science, 2001). No Brasil, o documento dos Grandes Desafios da SBC (SBC, 2006) reforça esse apoio no desafio 2: "Modelagem Computacional". Ainda no Brasil, eventos como o e-Science 2007 (Mattoso *et al*., 2007) levantam discussões importantes e reúnem grupos de pesquisa no país que trabalham sobre os problemas da área.

Sistemas de gerência de *workflows* científicos (SGWfC) foram recentemente desenvolvidos para apoiar a gerência da composição de processos e dados num fluxo que possa encadear as diversas análises de dados. O uso desses sistemas representa um avanço, porém, o apoio computacional ao experimento científico em larga escala encontra-se ainda incipiente (Gil *et al*. 2007, WORKS'07).

A motivação principal desse artigo está em identificar e discutir desafios necessários para prover apoio computacional ao desenvolvimento de ciência em larga escala. Mais especificamente, visamos a enfrentar os desafios na gerência dos recursos distribuídos inerentes ao desenvolvimento de ciência atual em áreas como bioinformática, engenharia do petróleo e a própria ciência da computação. Essa motivação está explicitamente identificada no desafio 2 dos Grandes Desafios da SBC (SBC, 2006), onde é dito que: "O objetivo deste desafio é criar, avaliar, modificar, compor, gerenciar e explorar modelos computacionais para todos esses domínios e aplicações". Pode-se dizer que experimentos científicos representam exemplos concretos para os modelos computacionais mencionados no desafio 2.

O restante deste texto está organizado como segue. A Seção 2 apresenta os desafios que identificamos para prover apoio computacional ao desenvolvimento de ciência em larga escala. A seção 3 apresenta um sumário e destaca pontos que consideramos importantes para atender aos desafios destacados na seção 2. Por último, a seção 4 apresenta as considerações finais.

2. Os Desafios do Apoio à Ciência em Larga Escala

Considerando que o principal recurso do desenvolvimento de ciência em larga escala é o experimento científico, nesse trabalho são analisados os diversos desenvolvimentos computacionais de apoio à experimentação. Para tanto, analisamos experimentos científicos *in-vitro*, *in-silico*, *in-virtuo*, além de processos experimentais em engenharia de software, simulações em larga escala, entre outros. Infra-estruturas computacionais para apoiar experimentação (Travassos *et al*., 2008) devem combinar processos de experimentação e de empacotamento de experimentos científicos, padrões de representação de dados, gerência de conhecimento, caracterização de ferramentas, serviços, modelos computacionais de qualidade e modelos necessários para apoiar experimentos em suas diversas etapas. Um dos desafios adicionais para a gerência de

experimentos científicos reside nos ambientes distribuídos tipicamente adotados na execução de vários desses experimentos. Vale destacar a dificuldade em realizar essa gerência em ambientes de grades computacionais (Pacitti *et al*., 2007).

Os desafios derivados que destacamos aqui lidam com o problema geral de gerência de experimentos científicos. Identificamos nesse problema três vertentes de pesquisa complementares na gerência do experimento científico, a saber: o **desempenho**, o **processo de gerência** e o **apoio semântico**.

O **desempenho** é uma das principais características do grande desafio 1: "Como lidar com grandes volumes de dados (SBC, 2006)?" Uma das técnicas mais consolidadas para tratar com grandes volumes de informação é a de dividir para conquistar. Nesse sentido, a fragmentação da informação e o processamento distribuído e paralelo sobre esses fragmentos é uma solução natural. O cenário atual tem como principal desafio gerar técnicas inovadoras de distribuição de dados e processamento paralelo para um ambiente dito legado. Cada vez mais os dados serão manipulados por sistemas pré-existentes ou específicos para formatos científicos, por exemplo. O gerenciamento e transparência de acesso a tais recursos em um ambiente altamente distribuído, com uma boa relação entre custo/desempenho, é um problema difícil. 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. O acesso a dados e processamento de consultas em grades é ainda um problema em aberto (Anjomshoaa *et al*., 2005). Experimentos com paralelismo em consultas OLAP em grades são apresentados em (Kotowski *et al.*, 2008). O acesso a grandes volumes de dados a partir da execução de um workflow torna o problema ainda maior. Vários SGWfC são voltados para a execução em grades visando ao desempenho, conforme levantamento em (Yu e Buyya, 2005). Porém, muitas vezes, apenas parte do workflow necessita de recursos em grade. A execução distribuída de um workflow envolve muitas vezes o uso de diferentes SGWfC, tornando a gerência complexa devido à heterogeneidade (Cruz *et al*. 2008a,b,c).

O **processo de gerência do experimento científico** passa por todo apoio ao ciclo de vida do experimento, que pode ser agrupado em duas principais etapas: concepção e execução.

Em relação à concepção, o problema consiste em definir novos processos de elicitação de requisitos junto à área científica, aplicando engenharia de software em outros domínios científicos, como ressaltado no grande desafio 2: Modelagem computacional de sistemas complexos artificiais, naturais e socioculturais e da interação homem-natureza (SBC, 2006). Na grande maioria das vezes, um experimento científico é concebido a partir de componentes pré-existentes (Wroe *et al.* 2007). Considerando que linguagens de *workflow* são utilizadas para encadear esses componentes, uma abordagem promissora está na extensão de técnicas de Reutilização de Software para o processo de modelagem do experimento. A Reutilização de Software é a disciplina responsável pela criação de sistemas de software a partir de software preexistente (Krueger, 1992). De forma análoga, é possível conceber *workflows* de experimentos científicos a partir de *workflows* preexistentes. Alguns trabalhos já vêm sendo desenvolvidos no sentido de aplicar Reutilização de Software para processos de software (Barreto *et al*., 2007a, 2007b). Esses trabalhos se fundamentam na premissa

que processos de software também podem se beneficiar das técnicas já existentes para o gerenciamento de software (Osterweil, 1987). O desafio está em estender essas técnicas e aplicá-las no contexto de *workflows* científicos, visto que *workflows* científicos guardam grande similaridade com processos de software. Ainda em relação à etapa de concepção, a forma de elicitação pode ser aprimorada com a adoção de características de agilidade apresentada pelos Métodos Ágeis (Beck *et al,* 2001; Abrantes e Travassos, 2007) para a aplicação nesse contexto, visando a uma modelagem iterativa e incremental dos *workflows* científicos, aliada à reutilização de fragmentos de *workflows* previamente existentes.

Em relação à execução do experimento, técnicas inovadoras que vêm sendo propostas para experimentação e gerência do conhecimento científico em Engenharia de Software (Travassos e Barros, 2003; Shull *et al*., 2004) também poderiam ser repensadas para o contexto científico como um todo. Outros desafios surgem ao registrar as atividades envolvidas no uso dos recursos científicos em ambientes como *grids*, de modo a obter controle e escalabilidade na gerência desses experimentos. Finalmente, técnicas utilizadas em depuração (*debugging*) de software poderiam ser adaptadas para o gerenciamento de execuções interativas de *workflows* científicos, possibilitando o acompanhamento e a análise parcial dos resultados em pontos predeterminados (i.e., *breakpoints*).

Já do lado **semântico**, a pesquisa que vem sendo realizada em proveniência de dados e processos tem sido apontada como promissora para tratar o principal objetivo do grande desafio 2, onde é dito que (SBC, 2006): "O objetivo deste desafio é criar, avaliar, modificar, compor, gerenciar e explorar modelos computacionais para todos esses domínios e aplicações.". Uma das tecnologias facilitadoras para todas essas ações é a proveniência, que associa semântica às diversas etapas do experimento científico. Uma vez registrada a proveniência, fica mais simples criar, avaliar ou modificar os modelos computacionais (ou experimentos científicos) com base no conhecimento acumulado do que vem sendo feito. A área de banco de dados vem tratando a semântica de dados há décadas. Diversos esforços associando o uso de ontologias junto a SGWfC vêm sendo realizados (Daltio e Medeiros, 2007; Digiampetri *et al.* 2007), cabendo um destaque ao pioneirismo do projeto myGrid (Wolstencroft *et al.* 2007). Já a área de Engenharia de Software vem tratando a semântica de processos também há décadas. Entretanto, num experimento científico esses dois conceitos estão intimamente entrelaçados. É necessário uma interseção forte entre essas duas áreas para tratar de modo adequado o desafio que envolve dados e processos de experimentos científicos. Para tanto, é possível associar tecnologias do estado da arte para gerar inovações na gerência de recursos científicos. Por exemplo, trabalhos em Gerência de Configuração (Murta, 2006) e Rastreabilidade (Murta *et al*., 2008), pertencentes à área de Engenharia de Software, tratam de questões relacionadas ao controle da evolução de software e fornecimento da razão de existência de artefatos gerados a partir de transformações consecutivas do software. Tais conceitos têm possivelmente muito a contribuir para a área de proveniência em *workflows* científicos, uma vez que a proveniência passa pela rastreabilidade (Provenance Challenge 2007), e versões de um mesmo *workflow* poderiam ser vistas como configurações de software gerenciáveis. Trabalhos recentes de proveniência em *workflows* científicos (Stevens et al., 2007; Davidson *et al.* 2007; Oinn *et al*., 2006; Anderson *et al*., 2007) possuem muita sofisticação na modelagem da

proveniência, porém, parecem ignorar todo o desenvolvimento na área da engenharia de software, adotando técnicas simples tanto para versões quanto para a rastreabilidade.

Dado este cenário, identificamos dois desafios derivados importantes: (i) gerência de experimentos científicos, envolvendo especificação, instanciação, execução e gerência de *workflows*, e (ii) proveniência em *workflows* científicos. Estes desafios são detalhados nas subseções a seguir, à luz da integração de bancos de dados e engenharia de software.

2.1. Gerência de Experimentos Científicos

Atualmente, cientistas das mais diversas áreas encontram apoio para suas pesquisas em um vasto conjunto de ferramentas computacionais para acelerar seu processo de análise e descoberta. Essas tecnologias permitem que grupos de pesquisadores colaborem com dados, idéias e experimentos em tempo real, mesmo quando estão geograficamente distribuídos. Essa colaboração envolve o uso de ferramentas variadas para cada tarefa do processo científico. Surge, então, a necessidade de orquestrar as soluções computacionais utilizadas para que a integração dos resultados gerados ao longo do processo não seja uma tarefa manual executada pelo cientista. Além disso, esta integração é necessária para que os dados produzidos por cada etapa do experimento possam permitir análises dessas informações.

Um primeiro desafio reside na concepção do *workflow*. A simples redefinição de workflows científicos por meio de uma linguagem estruturada para SGWfC já aumenta a qualidade e confiança dos resultados (Cavalcanti *et al.* 2002, Cruz *et al.* 2008d). Algumas características de agilidade presentes nos Métodos Ágeis podem ser adaptadas para o contexto em questão, visando uma concepção iterativa e incremental do *workflow* científico. Para isso, um desafio está em como conceber uma versão inicial do *workflow* em curto espaço de tempo e refinar essa versão, incrementalmente, até que este tenha todas as funcionalidades necessárias, sem que funcionalidades adicionais, mas não necessárias, tornem o mesmo complexo. Espera-se, com este procedimento, permitir a garantia da qualidade do workflow, detectando e resolvendo os problemas de forma pró-ativa, evitando que se incorra em retrabalho.

Por outro lado, durante a concepção de um *workflow* científico são identificadas oportunidades de reutilização de programas existentes ou até mesmo de fragmentos de *workflows* previamente concebidos. Nessas situações, outro desafio consiste na adoção de técnicas de reutilização de software e gerência do conhecimento científico (Mendonça *et al*., 2008) para sistematizar o processo de busca, recuperação e adaptação de componentes reutilizáveis.

Uma vez definido o *workflow* que modela o experimento científico, é necessário personalizá-lo, executá-lo dando prosseguimento às diversas etapas do experimento. Inúmeros SGWfC estão disponíveis para gerenciar a execução de experimentos científicos (Akram *et al*., 2006; Altintas *et al*., 2004, 2005, 2006; Callahan *et al*., 2006; Cavalcanti *et al*., 2005; Foster *et al*., 2002, 2003; Stevens *et al*., 2007; Goble e DeRoure, 2007; Ludäscher *et al*., 2006; Oinn *et al*. 2006; Anderson *et al*. 2007; Ptolemy Project, 2007; Taverna Workbench, 2007; Venugopal *et al*., 2006; Yu e Buyya, 2005; Zanikolas e Sakellariou, 2005). Cada sistema, embora genérico, tem suas características de destaque, às vezes focado em um domínio de aplicação. Usando interfaces gráficas que

permitem a modelagem e o acompanhamento da execução de experimentos científicos, os cientistas podem encadear, de forma controlada, diversas ferramentas.

Esse grande número de opções é justamente um dos problemas, já que cada um possui suas especificidades, embora tendam a ser genéricos. Cada um desses sistemas possui um modelo próprio de representação de *workflows*, uma linguagem própria de especificação e uma máquina própria de execução de *workflows*. Esse cenário dificulta a mudança do SGWfC, ou a possibilidade de usar uma outra máquina de execução, ou mesmo um outro ambiente de execução. Visando à colaboração entre cientistas na realização de experimentos, o projeto myExperiment propõe o uso de redes sociais num ambiente de pesquisa virtual (myExperiment 2008).

Entretanto, o desafio principal aqui é prover uma camada de software que permita a gerência de experimentos científicos de modo independente de um SGWfC. É preciso definir e desenvolver serviços que possam ser aplicados a SGWfC já existentes e bem sucedidos. O fio condutor deste desafio reside em trabalhar o máximo possível com os poucos padrões existentes (van der Aalst, 2004), um formalismo para descrição do *workflow* (Braghetto *et al*., 2007) e modelos genéricos a serem mapeados para sistemas pré-existentes. É importante oferecer recursos para a geração de heurísticas que permitam a geração de um plano de execução paralela eficiente para um *workflow* científico, independente de máquina de execução do *workflow* ou do ambiente distribuído. A idéia é prover aos usuários não especialistas em processamento paralelo, uma solução para a execução de seus *workflows* com melhor desempenho, independente da máquina de execução escolhida. Pesquisas nessa direção estão sendo executadas (Cruz *et al*., 2008c).

Aglomerados (*Clusters*) de PCs, *grids* e a *Web* vêm sendo utilizadas como plataformas para a execução de programas científicos e *workflows*. A execução eficiente de *workflows* é baseada na escolha dos melhores nós para execução, e na troca de dados entre os nós. Existem diferentes alternativas para executar um *workflow* científico em paralelo, já que os programas e dados podem ser distribuídos de diferentes maneiras. A escolha da melhor estratégia deve levar em consideração as dependências entre os componentes, os diferentes tempos de execução dos programas e os diferentes tamanhos dos arquivos processados. Alguns desses aspectos foram avaliados no contexto de *workflows* em grades (Meyer *et al*., 2006a, 2006b).

Técnicas típicas de otimização de consultas não podem ser aplicadas, uma vez que (i) o conjunto de nós participantes não é conhecido a priori, (ii) chamadas de serviços ou programas podem resultar em outras chamadas – isso proíbe a utilização da estratégia de programação dinâmica, e (iii) devido à volatilidade dos nós ou pontos da rede, um plano computado em um dado momento pode se tornar inválido até a hora de sua execução (Pacitti *et al*., 2007). Diante disso, é necessário desenvolver uma estratégia dinâmica de otimização baseada em custos. O núcleo desta estratégia seria um algoritmo adaptativo que busca planos que sejam bons o suficiente, priorizando o processamento paralelo, com custos baixos de comunicação. Algumas iniciativas nessa direção vêm sendo desenvolvidas e avaliadas (Ruberg e Mattoso, 2008; Pereira *et al*., 2006). Entretanto, os aspectos dinâmicos e heterogêneos nesses ambientes deixam vários problemas em aberto.

Finalmente, técnicas utilizadas em depuração de software precisam ser adaptadas para o gerenciamento de execuções interativas de *workflows* científicos, possibilitando o acompanhamento e a análise parcial dos resultados em pontos predeterminados. Esta característica é extremamente necessária aos cientistas. No entanto, poucos SGWfC oferecem tal apoio.

2.2. Proveniência de dados e processos

Aplicações científicas se caracterizam por possuírem grandes quantidades de dados complexos. A independência do desenvolvimento dessas aplicações gera um alto grau de heterogeneidade na representação desses dados. Na medida em que aumentam o armazenamento de formatos heterogêneos de dados, a natureza distribuída dos *workflows* e a necessidade de prover um caminho integrado para visualizar dados das diferentes etapas do experimento, determinadas questões se tornam cada vez mais importantes. São elas: Quando se deve capturar ou armazenar o dado? Qual dado deve ser incluído? Onde e como armazenar esse dado?

Mesmo decidindo a questão de armazenamento dos dados, surgem os aspectos de proveniência associada aos experimentos. Um dos objetivos é obter uma base de experimentos e preservar a memória corporativa científica. A modelagem da proveniência visa responder a perguntas genéricas sobre o experimento, como, por exemplo: Que análises estão disponíveis? Como unificar e resumir os conhecimentos gerados? Como consultar uma base de Experimentos? Quais projetos envolvem de alguma forma dados com a característica X? Quais experimentos usam o programa P independente da versão? Ou ainda visa responder questões mais específicas, como: De onde veio esse dado? De onde veio essa imagem? Com que parâmetros ela foi gerada? Qual a diferença no processo utilizado para gerar essas duas imagens? Que versão do programa P foi usada para executar o *workflow* Y?

Dentre os SGWfC que já oferecem algum apoio à gerência de dados de proveniência, é possível destacar o myGrid (Stevens *et al*., 2007), o Kepler (Altintas *et al*., 2004) e o Vistrails (Callahan *et al*., 2006). Esses sistemas participaram dos diversos *Provenance Challenge* (2007) e possuem mecanismos distintos para gerenciar os dados envolvidos nos *workflows* modelados. Entretanto, se parte do *workflow* precisa ser executada numa grade computacional ou numa máquina paralela, cabe ao SGWfC da grade gerenciá-lo. Um dos desafios é manter a captura da proveniência independente da máquina de execução. Resultados iniciais relacionados a esta questão já podem ser observados (Cruz *et. al*., 2008a).

A gerência dos recursos, em especial dos dados envolvidos nos experimentos científicos, é uma tarefa árdua para os SGWfC, pois cada domínio de aplicação científica, como bioinformática ou geologia, por exemplo, possui esquemas, metadados ou ontologias mais adequados ao seu contexto. Os esquemas de dados propostos pelos SGWfC são, de um modo geral, extremamente simples e não substituem os esquemas de dados dos domínios de aplicação que, por sua vez, não são facilmente incorporados aos SGWfC (Cruz *et al*., 2008b).

Uma deficiência ainda comum para estes SGWfC é a falta de interfaces adequadas para a construção de consultas aos dados armazenados durante a execução. O Vistrails definiu uma linguagem de consulta chamada *Vistrails Query Language*

(Anderson *et al*., 2007) para possibilitar consultas a evoluções de *workflows* e informações armazenadas durante a execução, mas ainda não provê uma interface para construção facilitada de consultas.

Uma possível direção de pesquisa para o desafio aqui é responder às questões expostas nesta seção por meio da aplicação de técnicas de Gerência de Configuração para a proveniência de dados em *workflows* científicos. Neste contexto, um dos componentes da Gerência de Configuração, o controle de versões, pode ser aplicado para identificar inequivocamente, em diferentes momentos, as configurações de um determinado *workflow* científico, assim como os dados produzidos por esse *workflow*. É necessário integrar esse conhecimento a um SGWfC com serviços de armazenamento de dados e de proveniência, para possibilitar a visualização dos dados de proveniência e dados intermediários gerados por um experimento, assim como realizar consultas analíticas sobre a massa de dados acumulada ao longo das diversas execuções de *workflows* efetuadas durante as pesquisas do experimento científico.

3. Sumário

Considerando todo o cenário descrito anteriormente, nesta seção apresentamos um resumo dos pontos nos quais acreditamos que as soluções poderão trazer ganhos. Os pontos estão categorizados de acordo com momentos distintos do ciclo de vida de um experimento científico, que podem englobar parte dos dois desafios derivados discutidos na Seção 2. São eles:

Concepção de *workflows*:

- Permitir aos cientistas e pesquisadores trabalharem nas aplicações chave do experimento de forma coordenada, sistematizada e com reutilização de conhecimento científico.
- Apoiar a modelagem e registro do conhecimento do domínio usando ontologias.
- Apoiar a modelagem dos protocolos de experimento, envolvendo a utilização das ontologias e dos *workflows* científicos abstratos e concretos.
- Permitir a definição de pacotes de laboratório, descrevendo os experimentos e o cadastro dos diversos recursos científicos envolvidos num experimento, como por exemplo: modelo científico; algoritmos, programas, versões de programas, *workflows*, versões de *workflows* e bases de dados. Esse cadastro é realizado uma única vez sendo guiado por ontologias previamente definidas. Assim, os recursos já estariam associados a diversas informações semânticas (por ex. por meio de inferências sobre as ontologias) e relacionados entre si (por ex. esse programa só pode ser executado com dados do tipo seg-Y).

Execução de *workflows*:

- Registrar as atividades desenvolvidas ao longo do processo do experimento científico. Esse registro poderá ser automático e usufruir da "documentação" dos recursos científicos cadastrados na infra-estrutura.
- Prover a execução eficiente dos *workflows* científicos do experimento, com foco em ambientes distribuídos e em grades.

Análise de *workflows*:

- Prover a comparação de resultados de visualização de saídas de execução dos *workflows* científicos por meio da referência direta à atividade do *workflow* que gerou a imagem, ou seja, a partir das imagens de visualização de resultados, pode-se consultar todos os programas que foram utilizados e seus respectivos parâmetros.
- Permitir consultas *ad-hoc* aos registros dos experimentos de forma a obter a proveniência dos dados, dos processos e de conhecimento.

Desta forma, quando combinados, esses pontos podem contribuir para o aumento da sistematização do ciclo de vida do experimento científico, desde a sua concepção até a sua execução e análise, favorecendo um maior entendimento dos resultados obtidos e viabilizando uma possível reprodução futura do experimento. Estes objetivos, quando atendidos, colaboram para a solução dos problemas do Grande Desafio número 2 da SBC. Para atendê-los, de forma diferenciada do que vem sendo proposto nos diversos projetos de pesquisa do estado da arte referenciados anteriormente, pretende-se aplicar conhecimentos adquiridos em pesquisas nas áreas envolvidas (banco de dados e engenharia de software), de modo que técnicas de uma área ajudem a resolver problemas de outra, sempre que possível. Mais especificamente, nossa experiência nas áreas de reutilização de software (Werner, 1992a, 1992b; Vasconcelos e Werner, 1997, 1998;), gerência de configuração de software (Murta, 2006; Murta *et al*., 2007; Murta *et al*., 2008), experimentação e gerência do conhecimento científico (Travassos e Barros, 2003; Shull *et. al*., 2004; Mendonça *et al*., 2008), gerência de *workflows* científicos (Cavalcanti *et al*., 2005; Targino *et al*., 2005; Cruz *et al*., 2008a, b, c; Dávila *et al*., 2005, 2008), otimização de chamadas de serviços em redes P2P (Pereira *et al*., 2006; Ruberg e Mattoso, 2008;), processamento de consultas em ambientes distribuídos (Mattoso *et al*., 2005; Ferraz *et al*., 2007; Figueiredo *et al*., 2007; Paes *et al*., 2008; Kotowksi *et al.* 2008) entre outras, contribuirão para a solução dos desafios levantados neste trabalho.

4. Considerações Finais

Este artigo apresentou um levantamento inicial e estratégias de solução de desafios derivados a serem vencidos para alcançar um objetivo maior: atender ao Grande Desafio número 2 da SBC. Este levantamento inicial faz parte de um projeto de pesquisa intitulado "GExp: Gerência de experimentos científicos em larga escala", aprovado pelo Edital MCT/CNPq/CT-INFO nº 07/2007. Neste projeto, pretendemos detalhar e implementar as propostas para cada um dos desafios derivados levantados neste artigo.

O projeto conta com vários especialistas nas diversas áreas do conhecimento envolvidas, tais como banco de dados, gerência de configuração, reutilização de software e engenharia de software experimental. Mais especificamente, estão envolvidos 3 pesquisadores CNPq, recém-doutores, além de especialistas internacionais. O projeto conta também com a participação de 3 alunos de doutorado, 7 de mestrado e 3 de graduação. Mais detalhes sobre o projeto podem ser encontrados em http://gexp.nacad.ufrj.br.

Alguns resultados concretos já foram obtidos (Cruz *et al*., 2008a, b, c; Oliveira *et al.*, 2008). O trabalho de Oliveira *et al*. (2008) foca em projeto de *workflow*. Eles propõem um serviço de recomendação que sugere combinações freqüentes de programas científicos para reutilização em *workflows*. O serviço é baseado em técnicas de reutilização de componentes e mineração de dados. Um protótipo foi implementado usando a máquina de gerenciamento de *workflows* do Vistrails. Já Cruz *et al*. (2008 c) também trata de projeto de *workflows*, mas seu foco está no controle de execução paralela. Eles propõem um *midleware* que monitora o projeto e controle de atividades paralelas em um *workflow* científico distribuído.

O problema de proveniência em ambientes distribuídos também já foi abordado. Cruz *et al*. (2008a) apresentam uma arquitetura de serviços que captura e armazena dados de proveniência de recursos distribuídos autônomos, replicados e heterogêneos. Tais dados podem ser usados para rastrear o histórico do processo de execução distribuído de modo independente.

Cruz *et al*. (2008b) tentam minimizar o problema de heterogeneidade das linguagens de definição de *workflow* e de seus ambientes de execução. Esta heterogeneidade causa problemas quando se quer, por exemplo, executar parte de um *workflow* em outro SGWfC para prover paralelismo. Neste caso, é necessário fazer um mapeamento entre as linguagens aceitas por cada um dos SGWfC envolvidos antes de prosseguir com a execução. Este problema pode ser minimizado através do uso de estruturas de controle independentes. Quando se insere estes módulos de controle em um *workflow*, o controle de execução fica menos dependente da máquina de execução do SGWfC.

Os trabalhos acima foram avaliados em diferentes SGWfC (Vistrails, Kepler e Taverna). Dada a diversidade dos SGWfC e a conseqüente falta de padrões, não consideramos a possibilidade de construção de mais um SGWfC. O objetivo é o desenvolvimento de um ferramental que possa ser utilizado para a concepção do experimento, de forma independente do SGWfC, utilizando-se técnicas de Engenharia de Software. Já para o apoio à execução paralela e distribuída do *workflow*, a proposta é construir módulos complementares a estes sistemas para enriquecer tanto a semântica do workflow quanto a execução, nos moldes do que já iniciado por Cruz *et al.* (2008b, 2008c). Estas extensões serão avaliadas experimentalmente em ambientes reais, em projetos de bioinformática e prospecção de petróleo que envolvem parceiros da Fiocruz, Biofísica/UFRJ, EELA e Petrobrás.

Referências

Abrantes, J.F.; Travassos, G.H. (2007). *Revisão quasi-Sistemática da Literatura: Caracterização de Métodos Ágeis de Desenvolvimento de Software*. Relatório Técnico ES-714/07. COPPE/UFRJ. http://www.cos.ufrj.br/uploadfiles/1196073281.pdf. (último acesso em 06/05/2008).

Akram, A.; Meredith, D.; Allan, R. (2006). *Evaluation of BPEL to Scientific Workflows*. In: CCGRID. May, vol. 1, pp. 269- 274.

Altintas, I.; Barney, O.; Jaeger-Frank, E. (2006). *Provenance Collection Support in Kepler Scientific Workflow System*. In: International Provenance and Annotation Workshop, pp. 1-15.

Altintas, I.; Berkley, C.; Jaeger, E.; Jones, M.; Ludäscher, B.; Mock, S. (2004). *Kepler: An Extensible System for Design and Execution of Scientific Workflows*. In: SSDBM, June, pp. 423- 424.

Altintas, I.; Birnbaum, A.; Baldridge, K.; Sudholt, W.; Miller, M.; Amoreira, C.; Potier, Y.; Ludaescher, B. (2005). *A Framework for the Design and Reuse of Grid Workflows*. In: International Workshop on Scientific Applications on Grid Computing (SAG), LNCS 3458, Springer, pp. 120-133.

Anderson, E.; Callahan, S.; Freire, J.; Koop, D.; Santos, E.; Scheidegger, C.; Silva, C.; Smith, N.; Vo, H. (2007). *Provenance Challenge - Vistrails*, disponível em http://twiki.ipaw.info/bin/view/Challenge/VisTrails, consultado em Maio de 2007.

Anjomshoaa, A.; Antonioletti, M.; Atkinson, M.; Baxter, R.; Borley, A.; Hong, N.; Collins, B.; Hardman, N.; Hicken, G.; Hume, A.; Knox, A.; Jackson, M.; Krause, A.; Laws, S.; Magowan, J.; Palansuriya, C.; Paton, N.; Pearson, D.; Sugden, T.; Watson, P.; Westhead, M. (2005) The Design and Implementation of Grid Database Services in OGSA-DAI. *Concurrency and Computation: Practice and Experience*, v. 17(2-4), pp. 357-376.

Barreto, A. S.; Rocha, A. R. C.; Murta, L. G. P. (2007). *Uma Abordagem Baseada em Técnicas de Reutilização para a Definição de Processos de Software*. In: SBQS, Workshop de Teses e Dissertações em Qualidade de Software (WTDQS), Porto de Galinhas.

Barreto, A. S.; Rocha, A. R. C.; Murta, L. G. P., (2007). *Uma Abordagem de Definição de Processos de Software Baseada em Reutilização*. Workshop de Implementadores MPS.BR, Belo Horizonte, pp. 33-39.

Beck, K.; Beedle, M.; Van Bennekum, A.; et al. (2001). *Manifesto for Agile Software Development*. Disponível em http://agilemanifesto.org, acessado em 27/03/2008.

Bowers, S. (2007). *Accelerating Scientific Knowledge Discovery through Scientific Workflows*. In: International Conference on Business Process Management, Keynote Speaker. Disponível em http://bpm07.fit.qut.edu.au/keynotes/shawn_bowers/bowers.pdf, acessado em 27/03/2008.

Braghetto, K. R.; Ferreira, J. E.; Pu, C. (2007). *Using Control-Flow Patterns for Specifying Business Processes in Cooperative Environments*. In: ACM SAC, 2007, Seoul, v. 2. pp. 1234-1241.

Callahan, S. P.; Freire, J.; Santos, E.; Scheidegger, C.E.; Silva, C.T.; Vo, H.T. (2006). *Vistrails: Visualization meets Data Management*. In: Proceedings of ACM SIGMOD, pp. 745-747.

Cavalcanti, M. C.; Targino, R.; Baião, F.; Rossle, S.; Bisch, P.; Pires, P. F.; Campos, M. L. M.; Mattoso, M. L. Q. (2005). Managing Structural Genomic Workflows using Web Services. *Data & Knowledge Engineering*, Elsevier, v. 53(1), p. 45-74.

Cavalcanti, M. C.; Mattoso, M.L.Q.; Campos M.L.; Llirbat F.; Simon E. (2002). *An Architecture for Managing Distributed Scientific Resources.* In: SSDM, IEEE Press, Edimburgo, Escócia, pp. 47-47.

Cruz, S.M.S.; Barros, P.; Bisch, P.; Campos, M.L.M.; Mattoso, M.L.Q. (2008a). *Provenance services for distributed workflows*. In: IEEE CCGrid, Lyon, pp. 526-533.

Cruz, S.M.S; Chirigati, F.S;Dahis, R.; Campos, M.L.M; Mattoso, M. (2008b). *Using explicit control processes in distributed workflows to gather provenance.* In: Second International Provenance and Annotation Workshop, IPAW 2008, Utah, EUA, a ser publicado na série LNCS.

Cruz, S.M.S.; Silva, F.N.; Gadelha Jr., L.M.R.; Cavalcanti, M.C.; Campos, M.L.M.; Mattoso, M.L.Q. (2008c). *A Lightweight Middleware Monitor for Distributed Scientific Workflows*. International Workshop on Workflow Systems in e-Science. In: IEEE CCGrid, Lyon, pp. 693-698.

Cruz, S.M.S.; Silva, E.; Oliveira, F.T.; Vilela, C.; Cuadrat, R.R.C.; Dávila, A.M.R.; Campos, M.L.M.; Mattoso, M.L.Q. (2008d). *OrthoSearch: A Scientific Workflow Approach to Detect Distant Homologies on Protozoans*. In: ACM SAC, Fortaleza, v. II. pp. 1281-1285.

Daltio, J.; Medeiros, C.B. (2007). *Um Serviço de Ontologias para Sistemas de Biodiversidade*. In: SEMISH, pp.2143-2157.

Davidson, S.; Cohen-Boulakia, S.; Eyal, A.; Bertram Ludascher, Timothy McPhillips, Shawn Bowers, and Juliana Freire (2007). Provenance in Scientific Workflow Systems, *IEEE Data Engineering Bulletin*, 32(4), pp. 44-50.

Dávila, A.M.R.; Lorenzini, D.; Mendes, P.; Satake, T.; Sousa, G.; Campos, L.; Mazzoni, C., Wagner, G.; Pires, P.; Grisard, E.; Cavalcanti, M.; Campos, M.L.M. (2005). GARSA: genomic analysis resources for sequence annotation. *Bioinformatics*, v. 21(23), pp. 4302-4303.

Dávila, A. M. R.; Mendes, P; Wagner, G.; Tschoeke, D.; Cuadrat, R.; Liberman, F.; Matos, L.; Satake, T.; Ocaña, K.; Triana, O.; Cruz, S.; Jucá, H.; Cury, J.; Silva, F.; Geronimo, G.; Ruiz, M.; Ruback, E.; Silva, F.; Probst, C.; Grisard, E.; Krieger, M.; Goldenberg, S.; Cavalcanti, M.; Moraes, M.; Campos, M.; Mattoso, M. (2008). ProtozoaDB: dynamic visualization and exploration of protozoan genomes. *Nucleic Acids Research (Database Issue)*, v. 36, pp. 547-552.

Digiampietri, L.A.; Pérez-Alcázar, J.J.; Medeiros, C.B.(2007). An ontology-based framework for bioinformatics workflows. *IJBRA, Inderscience*, v. 3(3) pp. 268-285.

Ferraz, C.; Braganholo, V.; Mattoso, M. (2007). *Storing AXML documents with ARAXA*. In: SBBD, João Pessoa, PB, pp. 255-269.

Figueiredo, G.; Braganholo, V. ; Mattoso, M. (2007). *Um Mediador para o Processamento de Consultas sobre Bases XML Distribuídas*. In: Sessão de Demos do SBBD, 2007, João Pessoa, PB, pp. 21-26.

Foster, I.; Voeckler, J.; Wilde, M.; Zhao, Y. (2002). *Chimera: A Virtual Data System for Representing, Querying and Automating Data Derivation*. In: SSDM, Edinburgh, Scotland, pp. 37-46.

Foster, I.; Voeckler, J.; Wilde, M.; Zhao, Y. (2003). *The Virtual Data Grid: A New Model and Architecture for Data-Intensive Collaboration*. In: Conference on Innovative Data System Research (CIDR), Asilomar, CA, USA, pp. 11-11.

Gil, Y.; Deelman, E.; Mark H. Ellisman, Thomas Fahringer, Geoffrey Fox, Dennis Gannon, Carole A. Goble, Miron Livny, Luc Moreau, Jim Myers. (2007). Examining the Challenges of Scientific Workflows. *IEEE Computer*, v. 40(12), pp. 24-32.

Goble, C.; De Roure, D. (2007). *myExperiment: social networking for workflow-using e-scientists*. In: WORKS, Monterey, California, USA. http://eprints.ecs.soton.ac.uk/15095/

Kotowski, N.; Lima, A. A.; Pacitti, E.; Valduriez, P.; Mattoso, M. L. Q. (2008). Parallel Query Processing for OLAP in Grids. *Concurrency and Computation: Practice & Experience*, v. online, a ser publicado.

Krueger, C.W. (1992). Software Reuse. *ACM Computing Surveys*, v. 24(2), pp. 131-183.

Ludäscher, B.; Altintas, I.; Berkley, C.; Higgins, D.; Jaeger, E.; Jones, M.; Lee, E.; Tao, J.; Zhao, Y. (2006). Scientific workflow management and the Kepler system. *Concurrency and Computation: Practice and Experience*, v. 18(10), pp. 1039-1065.

Mattoso, M. L. Q.; Zimbrão, G.; Lima, A. A.; Baião, F.; Braganholo, V.; Aveleda, A. A.; Miranda, B.; Almentero, B. K. ; Costa, M. N. (2005). *ParGRES: Middleware para Processamento Paralelo de Consultas OLAP em Clusters de Banco de Dados*. In: SBBD - Sessão de Demos, Uberlândia. pp.19-24.

Mattoso, M. L. Q. (Org.); Ferreira, J.E. (Org.); Braganholo, V. (Org.). *E-Science Workshop*. Co-realizado ao SBBD/SBES. ed. Sociedade Brasileira de Computação, 2007. v. 1. 82 p.

Mendonça, M.G.; Maldonado, J.C.; Oliveira, M.C.; Carver, J.; Fabbri, S.; Shull, F.; Travassos, G.H., Höhn, E.N.; Basili, V. R. (2008). *A Framework for Software Engineering Experimental Replications*. In: IEEE International Conference on Engineering of Complex Computer Systems, Belfast, Northern Ireland, pp. 203-212.

Meyer, L. A. V. C.; Wilde, M.; Mattoso, M. L. Q.; Foster, I. (2006a). *Planning Spatial Workflows to Optimize Grid Performance*. In: ACM SAC, Dijon, v. 1. pp. 786-790.

Meyer, L. A. V. C.; Wilde, M.; Mattoso, M. L. Q.; Foster, I. (2006b). *An Opportunistic Algorithm for Scheduling Workflows on Grids*. In: Vecpar - LNCS, v. 4395. pp. 1-12.

Murta, L. G. P. (2006). *Gerência de Configuração no Desenvolvimento baseado em Componentes*, Leonardo Murta, Tese de Doutorado PESC/ COPPE/UFRJ, Rio de Janeiro, Outubro.

Murta, L. G. P.; Van Der Hoek, A.; Werner, C. M. L. (2008). Continuous and Automated Evolution of Architecture-to-Implementation Traceability Links. *Automated Software Engineering Journal*, v. 15, pp. 75-107.

Murta, L. G. P.; Oliveira, H. L. R.; Dantas, C. R.; Lopes, L. G. B.; Werner, C. M. L. (2007). Odyssey-SCM: An Integrated Software Configuration Management Infrastructure for UML models. *Science of Computer Programming*, v. 65 (3), pp. 249-274.

myExperiment (2008). *myExperiment Project*. Disponível em http://www.myexperiment.org/, acesso em 03/2008.

Oinn, T.; Greenwood, M.; Addis, et al. (2006). Taverna: Lessons in creating a workflow environment for the life sciences, *Concurrency and Computation: Practice & Experience*, v.18 (10), pp. 1067-1100.

Oliveira, F. T.; Murta, L.; Werner, C.; Mattoso, M. (2008). *Using Provenance to Improve Workflow Design*. In: International Provenance and Annotation Workshop, IPAW 2008, Utah, EUA, a ser publicado na série LNCS.

Osterweil, L. (1987), *Software Processes Are Software Too*. In: International Conference on Software Engineering, Monterey, Estados Unidos, Abril, pp. 2-13.

Pacitti, E.; Mattoso, M. L. Q.; Valduriez, P. (2007). Grid Data Management: Open Problems and New Issues. *Journal of Grid Computing*, v. 5(3), pp. 237-281.

Paes, M.; Lima, A. A.; Valduriez, P.; Mattoso, M.L.Q. (2008). *High-performance Query Processing of a Real-world OLAP Database with ParGRES*. In: VECPAR, Toulouse, a ser publicado.

Pereira, D.; Ruberg, G.; Mattoso, M.L.Q. (2006). *Geração Eficiente de Planos de Materialização para Documentos XML Ativos*. In: SBBD, Florianópolis. pp. 136-250.

Provenance Challenge (2007). Disponível em http://twiki.ipaw.info/bin/view/Challenge/WebHome, acesso em 03/2008.

Ptolemy Project (2007). Disponível em http://ptolemy.eecs.berkeley.edu/ptolemyII/, acesso em 05/ 2007.

Rajasekar, A.; Wan, M. (2002). *SRB & SRBRack – Components of a Virtual Data Grid Architecture.* Advanced Simulations Technologies Conference, San Diego, EUA.

Ruberg, G.; Mattoso, M.L.Q. (2008). *XCraft: Boosting the Performance of Active XML Materialization*. In: EDBT, Nantes, França, ACM Int. Conf. Proceeding Series, 2008. v. 261. pp. 299-310.

SBC (Sociedade Brasileira de Computação) (2006). *Grandes Desafios da Computação no Brasil: 2006-2016*. Disponível em http://www.sbc.org.br/index.php?language=1&content=downloads&id=272

Shull, F.; Mendonça, M.; Basili, V.; Carver, J.; Maldonado, J. C.; Fabbri, S.; Travassos, G. H.; Ferreira, M. C. (2004). Knowledge-Sharing Issues in Experimental Software Engineering. *Empirical Software Engineering*, v. 9(1-2), pp. 111-137.

Stevens, R.; Zhao, A.; Goble, C.A. (2007). Using provenance to manage knowledge of *In Silico* experiments. *Briefings in Bioinformatics*, v. 8(3), pp. 183-194.

Targino, R.; Cavalcanti, M. C.; Mattoso, M. L. Q. (2005). *An Environment to Define and Execute In-Silico Workflows Using Web Services*. In: International Workshop on Data Integration in the Life Sciences (DILS), San Diego. LNBI, v. 3615. pp. 288-291.

Taverna Workbench (2007). Disponível em http://taverna.sourceforge.net/index.php, acesso em 03/2008.

Travassos, G. H.; Barros, M. O. (2003). *Contributions of In Virtuo and In Silico Experiments for the Future of Empirical Studies in Software Engineering*. In: Workshop on Empirical Software Engineering: The Future of Empirical Studies in Software Engineering, Roma. Fraunhofer IRB Verlag, pp. 117-130.

Travassos, G.H.; Santos, P.S.M.; Mian, P.G.; Dias Neto, A.C.; Biolchini, J. (2008). *An Environment to Support Large Scale Experimentation in Software Engineering*. In: IEEE International Conference on Engineering of Complex Computer Systems (ICECCS). Belfast, Northern Ireland, pp. 193-202.

UK e-Science (2001). UK e-Science Programme. Disponível em http://www.rcuk.ac.uk/escience.

van der Aalst, W.M.P. (2004). *Business Process Management Demystified: A Tutorial on Models, Systems and Standards for Workflow Management*. LNCS, vol. 3098, pp. 1-65.

Vasconcelos Jr, F.; Werner, C. (1997). *Software Development Process Reuse based on Patterns*. In: ICSEKE, Madri, Espanha, junho, pp. 97-104.

Vasconcelos Jr, F.; Werner, C. (1998). Organizing the Software Development. Process Knowledge: An Approach Based on Patterns. *International Journal of Software Engineering & Knowledge Engineering*, v. 8(4), pp. 461-482.

Venugopal, S.; Buyya, R. ; Ramamohanarao, K. (2006). A Taxonomy of Data Grids for Distributed Data Sharing, Management, and Processing. *ACM Computing Surveys*, v. 38(1), Artigo 3, 53p.

Werner, C. (1992a). *Reutilização de Software no Desenvolvimento de Software Científico*, Tese de Doutorado do PESC/COPPE/UFRJ, Rio de Janeiro, Março.

Werner, C. (1992b). *Software Reusability in a Scientific Software Development Framework*. In: Computing in High Energy Physics conference, Annecy, França, 1992, pp. 579-582.

Wolstencroft, K.; Alper, P. ;Duncan Hull, Chris Wroe, Phillip W. Lord, Robert D. Stevens, Carole A. Goble. (2007). The myGrid ontology: bioinformatics service discovery. *IJBRA*, v. 3(3), pp. 303-325.

WORKS'07 (2007), *2nd Workshop on Workflows in Support of Large-Scale Science*, http://portal.acm.org/citation.cfm?id=1273360

Wroe, C.; Carole A. Goble, Antoon Goderis, Phillip W. Lord, Simon Miles, Juri Papay, Pinar Alper, Luc Moreau(2007). Recycling workflows and services through discovery and reuse. *Concurrency and Computation: Practice and Experience*, v. 19(2), pp. 181-194.

Yu, J.; Buyya, R. (2005). A taxonomy of scientific workflow systems for grid computing. *ACM SIGMOD Record*, vol. 34, pp. 44-49.

Zanikolas, S.; Sakellariou, R. (2005). A taxonomy of grid monitoring systems. *Future Generation Computer Systems,* v. 21, pp. 163–188.