Uma Arquitetura para Ecossistema de Software Científico

Vitor Freitas, José Maria N. David, Regina Braga, Fernanda Campos

Programa de Mestrado em Ciência da Computação (PGCC/DCC)
Universidade Federal de Juiz de Fora (UFJF) – Juiz de Fora, MG – Brazil
{vitor.freitas,jose.david, regina.braga, fernanda.campos}@ufjf.edu.br

Resumo. *A concepção de workflows científicos é uma abordagem utilizada no contexto de e-Science. Existem muitas pesquisas voltadas para o gerenciamento e execução de experimentos baseados em workflows. No entanto, experimentos complexos envolvem interações entre pesquisadores geograficamente distribuídos, demandando utilização de grandes volumes de dados, serviços e recursos computacionais distribuídos. Este cenário categoriza um ecossistema de experimentação científica. Para conduzir experimentos neste contexto, cientistas precisam de uma arquitetura flexível, extensível e escalável. Durante o processo de experimentação, informações valiosas podem ser perdidas e oportunidades de reutilização de recursos e serviços desperdiçadas, caso a arquitetura de ecossistema para e-Science não considere estes aspectos. Com o objetivo de tratar a extensibilidade de plataformas de ecossistemas, este trabalho apresenta uma arquitetura orientada a serviços apoiada por uma rede ponto a ponto, desenvolvida para tratar as etapas do ciclo de vida de um experimento científico. Este trabalho apresenta como contribuições uma arquitetura para ecossistemas de software científico, a implementação desta arquitetura, bem como a sua avaliação.*

Abstract. *The conception of scientific workflows is an approach used in the context of e-Science. There are many researches about the management and execution of experiments based on workflows. However, scientific experiments involve complex interactions between geographically distributed researchers, requiring the usage of large amount of data, services and distributed computing resources. This scenario categorizes a scientific experimentation ecosystem. In order to carry out experiments in this context, researchers need an architecture for e-Science that supports extensibility. During the experimentation process, valuable information can be unexploited and, as a result, reusing opportunities of resources and services could be lost if the ecosystem architecture for e-Science does not consider previous mentioned requirements. This works presents a service-oriented architecture supported by a peer-to-peer network. It was developed to support life-cycle stages of a scientific experiment. This work also presents, as contributions, an architecture to support experiments execution of scientific software ecosystems, as well as its evaluation.*

1. Introdução

Workflow científico é uma abordagem utilizada no contexto de e-Science e é relacionada a organização de um fluxo de execução de aplicações científicas, que

devem ser sequenciadas de forma a realizar o experimento. (Altintas et al., 2004). A concepção de *workflows* científicos não é uma tarefa trivial, demandando um conhecimento especializado, muitas vezes interdisciplinar, exigindo do cientista algum conhecimento em computação. Como resultado, criam-se algumas barreiras como, a dificuldade no desenvolvimento e, sobretudo, na reutilização de workflows concebidos por outros cientistas, levando muitas vezes ao retrabalho.

O conceito de Linha de Produtos de Software (LPS) vem sendo utilizado neste contexto (Castro et al., 2015). A utilização de uma LPS no contexto científico pode auxiliar cientistas na concepção e controle de *workflows*. No entanto, o processo de experimentação vai além da concepção. Experimentos complexos envolvem interações entre pesquisadores, utilização de grandes volumes de dados, de serviços e de recursos computacionais distribuídos, constituindo um ecossistema de software científico.

Muitas vezes um experimento científico é uma atividade colaborativa. Ele passa por um ciclo de vida que se inicia na investigação do problema, seguindo pela prototipação e execução do experimento, até finalmente chegar à etapa de publicação das contribuições (Belloum et al., 2011). Durante o processo de experimentação, informações podem ser perdidas e oportunidades de reutilização de recursos e serviços desperdiçadas, caso a arquitetura de suporte para e-Science não considere estes aspectos.

Neste contexto, este trabalho define um Ecossistema de Software Científico (ECOSC), denominado ECOS-PL Science, como um subconjunto de Ecossistema de Software (ECOS), conforme definido por Jansen et al. (2009). Um ECOSC pode ser definido pelas suas relações com fornecedores de software científico, institutos de pesquisa, pesquisadores, órgãos de fomento, instituições financiadoras, e as partes interessadas nos resultados de pesquisa. Portanto, a arquitetura de um ECOSC deve ser flexível, uma vez que ela pode integrar com plataformas científicas externas, que evoluem de maneira independente, e estão em constante evolução. Estes relacionamentos ocorrem para gerar maior valor para o ECOSC, os quais requerem a abertura de suas fronteiras onde aplicações terceiras passam a se conectarem e se beneficiarem de seus serviços, gerando valor para as partes envolvidas. Portanto, a arquitetura do ECOSC precisa ser extensível. Um ECOSC além de ser provedor de serviços, é também um consumidor de serviços de software científico, sendo necessário que sua arquitetura esteja apta a realizar novas integrações sem que haja modificações substanciais na arquitetura da solução. Finalmente, a arquitetura de um ECOSC precisa ser escalável, uma vez que ela é extensível, podendo ocasionar em um crescimento repentino e inesperado de requisições pelos serviços.

A proposta deste trabalho é se ter um ambiente compartilhado, que possibilite: (i) a presença simultânea de cientistas trabalhando em um mesmo experimento, (ii) o tratamento de grandes volumes de dados relativos ao processo de experimentação, (iii) a execução de *workfows* científicos dentro da plataforma, e (iv) a viabilização da evolução da plataforma do contexto de e-Science. No contexto deste artigo, apenas o requisito não funcional extensibilidade foi avaliado.

Este trabalho está dividido da seguinte forma. A seção 2 apresenta o conceito de experimentação científica e os principais trabalhos relacionados. A seção 3 discute a abordagem ECOS PL-Science, e apresenta a arquitetura proposta como solução. A

próxima seção (seção 4) descreve um estudo de caso e a aplicação de métricas para avaliação da extensibilidade da solução proposta. Finalmente, as considerações finais são apresentadas na seção 5.

2. Software Científico

Experimentos científicos complexos envolvem a utilização de dados e recursos computacionais distribuídos, demandando a colaboração de cientistas geograficamente distribuídos (Belloum et al., 2011). Com isso, uma nova geração de redes sociais de pesquisa surgem, como o myExperiment (Roure et al., 2009), proporcionando um ambiente colaborativo para descoberta, utilização e compartilhamento de *workflows* científicos. Belloum et al. (2011) descrevem o ciclo de vida de um experimento científico que se inicia na investigação do problema, seguindo pela prototipação e execução do experimento, até finalmente chegar à etapa de publicação das contribuições. Durante a etapa de investigação do problema é feita a busca por problemas relevantes, as ferramentas disponíveis são exploradas, objetivos são definidos e o problema é decomposto em etapas. Na etapa de prototipação do experimento o *workflow* é desenhado e os componentes necessários são desenvolvidos. Na etapa de execução do experimento ocorre a execução propriamente dita, controle, coleta e análise dos resultados. Finalmente os dados são anotados e as contribuições publicadas durante a etapa de publicação dos resultados.

Belloum et al. (2011) propõem um ciclo de vida do processo de experimentação científica colaborativa composto pelas etapas de investigação do problema, prototipação do experimento, execução e publicação dos resultados. Ainda que a colaboração e o compartilhamento de recursos ocorram, a abordagem não pode ser considerada um ECOSC, pois não atende aos requisitos de extensibilidade que uma plataforma de ECOSC demanda. Uma das ferramentas utilizadas pelo projeto com a finalidade de compartilhar arquivos faz uso de protocolos FTP (Grid-FTP e SSH-FTP), enquanto na proposta deste trabalho é implementada uma rede ponto a ponto, de modo a descentralizar o compartilhamento de arquivos. Outra diferença na proposta de Belloum et al. (2011) está na interoperabilidade dos recursos computacionais para apoiar as etapas do ciclo de vida de um experimento científico, e na ausência de um ambiente multiusuário.

A abordagem proposta por Mattoso et al. (2010) considera que um experimento científico passa por três etapas, tais como: composição, execução e análise. Ferramentas que dão suporte ao ciclo de vida de experimentação científico são discutidas, no entanto nenhuma integração ou ferramenta para apoiar todo processo é proposto. Zhang et al. (2014) propõem uma abordagem denominada Confucius, para o desenvolvimento de *workflows* científicos de maneira colaborativa, estendendo um gerenciador de *workflow* científico de código aberto. A abordagem tem o foco na atividade de composição de *workflows*, não tratando as etapas anteriores e posteriores do processo de experimentação. No mesmo sentido, a abordagem Co-Taverna é proposta por Zhang (2010), uma extensão do projeto Taverna. Novamente, a proposta não contempla todas as etapas de um experimento científico, embora contribua para o processo de experimentação colaborativo.

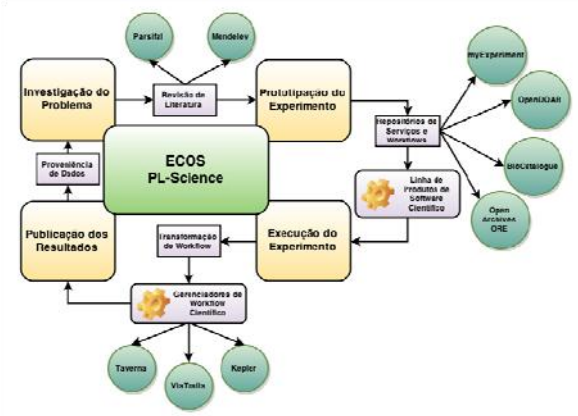

Figura 1. Ciclo de vida de um experimento científico no ECOS PL-Science

Mattoso et al. (2010) e Belloum et al. (2011) apresentam uma visão mais ampla do processo de experimentação, considerando as demais etapas de um experimento científico. Nenhum dos trabalhos relacionados utilizam uma abordagem de ECOS ou fazem uso de redes ponto a ponto, embora a proposta de Belloum et al. (2011) apresente alguns aspectos relacionados a ECOS.

3. ECOS PL-Science

Em sua essência, o ECOS PL-Science utiliza o conceito de ciclo de vida de um experimento científico proposto por Belloum et al. (2011). Explora cada etapa e provê recursos para auxiliar no processo de experimentação (Figura 1). O ciclo de vida foi adaptado, propondo a realização de revisões sistemáticas de literatura durante a etapa de investigação do problema e a utilização do conceito de uma LPSC na etapa de prototipação do experimento. Durante a etapa de prototipação são disponibilizados, aos cientistas, recursos para utilizarem *workflows* e serviços web de plataformas como myExperiment e BioCatalogue (Bhagat et al., 2010). A linha de produto de software científico Collaborative PL-Science é então incorporada na etapa de prototipação do experimento, aumentando o nível de reuso e a qualidade no desenvolvimento de *workflows* científicos, além de reduzir o tempo e consequentemente o custo do desenvolvimento. Finalmente, não somente os resultados da execução do experimento são armazenados no ECOS PL-Science, mas também todos os dados relativos ao processo de experimentação, possibilitando que outros pesquisadores possam consultar.

A arquitetura do ECOS PL-Science é apresentada na Figura 2. A camada de lógica de negócio é responsável pelo processamento, separando as responsabilidades pelo processamento dos dados da visualização, que neste ocorre via web. Com isto, a aplicação pode ser estendida para outras plataformas, como aplicações móveis por exemplo, sendo necessária somente a construção dos componentes de interface. A separação da visualização da lógica de negócio é importante para alcançar um nível maior de extensibilidade da plataforma, uma vez que a API se comunica com a camada de lógica de negócio para disponibilizar para aplicações externas, dados, funcionalidades e serviços da plataforma ECOS PL-Science.

A camada de visualização, ilustrada pela Figura 3 representa a aplicação web propriamente dita, onde os cientistas interagem com a aplicação em um ambiente multiusuário, executando atividades relativas a condução de um experimento científico.

A rede ponto a ponto trabalha diretamente no núcleo da aplicação, onde ocorre o gerenciamento dos artefatos da LPSC. Cada instância do ECOS PL-Science exerce o papel de um ponto na rede, compartilhando seus artefatos com outras aplicações, sendo estas conhecidas ou não. O núcleo da aplicação faz parte da proposta Collaborative PL-Science, onde elementos e serviços de colaboração foram associados à LPSC, onde estão representados os processos envolvidos na etapa de concepção de *workflows* científicos. A interação dos usuários externos ocorre em dois níveis, primeiro na camada de visualização da aplicação, atuando na condução de experimentos e no desenvolvimento de artefatos. O segundo nível ocorre no ambiente de desenvolvimento do ECOSC, gerenciado na plataforma GitHub. Através dela, desenvolvedores externos podem auxiliar na construção da plataforma, propondo melhorias e desenvolvendo novas funcionalidades. Essas funcionalidades serão avaliadas pela equipe de desenvolvimento interna da plataforma, podendo ou não serem integradas no código fonte. Cientistas ganham um canal de comunicação, através do qual podem solicitar novas funcionalidades ou reportarem problemas na plataforma.

Atualmente, o ECOS PL-Science está integrado com as plataformas Parsifal[1], Mendeley, Taverna Server (Zhang, 2010), myExperiment (Roure et al., 2009) e BioCatalogue (Bhagat et al., 2010). Por restrição de espaço, detalhes das integrações não são apresentados neste artigo[2]. No Com o desenvolvimento da proposta, conclui-se que são necessidades importantes para o sucesso de uma API: (i) a documentação completa de todos recursos disponíveis na API, (ii) a flexibilidade nos tipos de dados suportados, idealmente disponibilizar formatos XML e JSON, (iii) a coerência na implementação dos métodos HTTP (por exemplo, não utilizar método GET para recuperar dados e remover dados – neste caso método DELETE deveria ser utilizado), (iv) *sandbox*, ou um ambiente de desenvolvimento, para viabilizar os testes durante a integração, (v) a distribuição de software cliente de integração da API, (vi) estar engajado com a comunidade de desenvolvedores de código aberto, e (vii) dar suporte aos clientes de integração desenvolvidos por terceiros.

4. Avaliação da Solução Proposta

Um estudo de caso foi conduzido com o intuito de avaliar o requisito não funcional de extensibilidade da plataforma ECOS PL-Science. O escopo da avaliação foi definido com base no método GQM, descrito a seguir: **analisar** a arquitetura do ECOS PL-Science **com o propósito** de avaliar sua extensibilidade **sob o ponto de vista** dos desenvolvedores **no contexto** da evolução de um ECOSC. A partir do escopo, a questão de pesquisa foi definida: A arquitetura do ECOS PL-Science viabiliza a extensão de suas funcionalidades? A hipótese nula foi definida como: (**H0**) A arquitetura do ECOS PL-Science não viabiliza a extensão de suas funcionalidades. A hipótese alternativa foi definida como: (**H1**) A arquitetura do ECOS PL-Science viabiliza a extensão de suas funcionalidades. Com base na questão de pesquisa o estudo experimental foi planejado. A avaliação foi conduzida com um grupo de alunos de mestrado da UFJF cujos projetos de pesquisa estão diretamente ligados à evolução da plataforma ECOS PL-Science. Os participantes possuem conhecimento prévio da plataforma e começaram a atuar no

[1] http://parsif.al

[2] Maiores informações podem ser obtidas em http://pgcc.github.io/plscience.

desenvolvimento a partir da adoção de uma estratégia de ECOSC. O projeto tornou-se código aberto e é gerenciado pela plataforma GitHub.

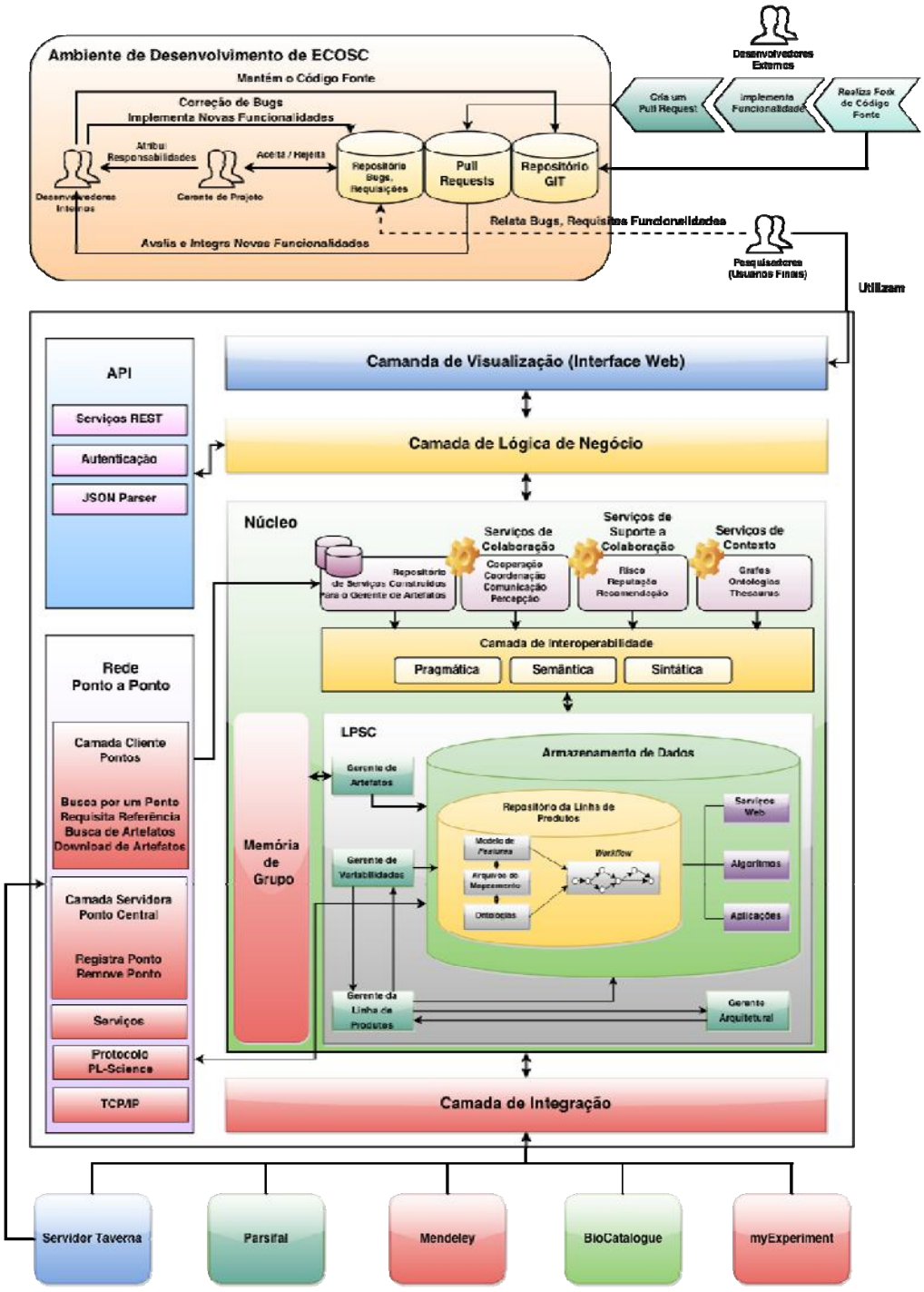

Figura 2. Arquitetura da Abordagem ECOS PL-Science

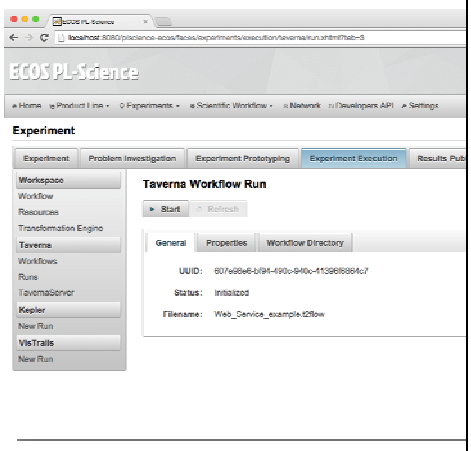

Figura 3. Interface Web ECOS PL-Science (visão parcial)

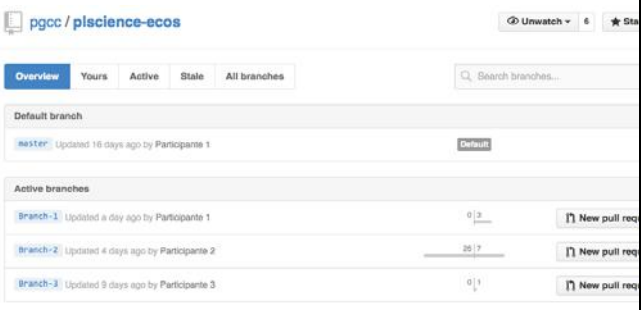

Figura 4. *Branches* na plataforma GitHub

Na plataforma GitHub, cada participante criou um *branch*, conforme pode ser visto na Figura 4 (os nomes dos participantes foram omitidos), a partir da versão mais recente e as funcionalidades começaram a ser desenvolvidas: (i) inclusão de elementos de colaboração e comunicação durante todas as etapas do experimento; (ii) integração da plataforma com ontologias de colaboração; (iii) suporte à interoperabilidade pragmática no desenvolvimento colaborativo de *workflows*. Os participantes foram orientados a se limitarem à implementação de suas funcionalidades, e que nesse momento não realizassem refatorações de código, por exemplo.

As implementações ocorreram em momentos distintos, e todas elas representam trabalhos em andamento que estendem a abordagem ECOS PL-Science. Os participantes foram entrevistados e como fonte de coleta de dados adicional, dados históricos do GitHub foram avaliados. Um dos indícios de que a arquitetura é extensível, é a quantidade de código fonte existente que foi alterado, para que novas funcionalidades fossem implementadas. O GitHub proporciona uma visão com base nos *commits*, de quantas linhas de código foram incluídas e quantas foram removidas. Um alto número de linhas removidas sugere que muitas linhas de código precisaram ser alteradas e adaptadas para que a funcionalidade fosse implementada. Os resultados da extração dos dados são apresentados na **Erro! A origem da referência não foi encontrada.**1.

Tabela 1. Extração de dados do estudo de caso

Participantes	Num. de Commits	Arq. Alterados ou Incluídos	Adições	Remoções
Participante 1	3	17	1951	24
Participante 2	7	80	16280	1
Participante 3	1	25	806	40

Através da análise dos dados a questão de pesquisa pode ser respondida. Comparando o número de remoções de linhas de código com o número de adições, pode-se concluir que a arquitetura do ECOS PL-Science pode ser estendida sem que haja grandes alterações de sua estrutura. Como resultado, existem evidências de que a hipótese nula (H0) pode ser rejeitada e a hipótese alternativa (H1) pode ser aceita.

5. Conclusões

Pode-se destacar como contribuição deste trabalho, o desenvolvimento de uma arquitetura para ECOSC para auxiliar cientistas na condução de experimentos colaborativos. Através desta arquitetura, plataformas de softwares científicos são integradas em um ambiente multiusuário, de modo a satisfazer às necessidades do processo de experimentação. Além disso, oferece suporte durante as etapas de investigação do problema, prototipação do experimento, execução e a publicação dos resultados no ciclo de vida de um experimento. Para tanto, um ciclo de vida foi estendido com o objetivo de apoiar experimentos científicos.

Durante o processo de desenvolvimento da plataforma, bem como durante a elicitação dos requisitos, uma dificuldade foi encontrar especialistas no domínio, de modo a alinhar a solução com necessidades reais do domínio. Outra questão é que a rede ponto a ponto desenvolvida neste trabalho é um protótipo, desenvolvida somente para avaliar a viabilidade de sua adoção em um ECOSC.

Um estudo de caso foi conduzido com o intuito de avaliar o requisito não funcional de extensibilidade da plataforma ECOS PL-Science. Os resultados obtidos foram promissores mas não podem ser generalizados, sendo válidos para o contexto do ECOS PL-Science. Em relação à escalabilidade e flexibilidade seria necessário uma quantidade maior de experimentos e, portanto, não foram considerados neste experimento. Esses requisitos não funcionais devem ser tratados em trabalhos futuros.

Referências Bibliográficas

Altintas, I., Berkley, C., Jaeger, E., Jones, M., Ludascher, B., Mock, S. (2004) “Kepler: an extensible system for design and execution of scientific workflows”, In: Scientific and Statistical Database Management. Proc. 16th Int.Conference on, p. 423–424.

Belloum, A. et al. (2011) “Collaborative e-Science Experiments and Scientific Workflows”, IEEE Computer Society vol.15, no. 4, pp. 39-47, doi:10.1109/MIC.2011.87.

Bhagat, J. et al. (2010) “BioCatalogue: A universal catalogue of web services for the life sciences”. Nucleic Acids Research, v. 38, p. 689–694.

Castro, G., Braga, R., David, J. M. N., Campos, F. (2015) “A Scientific Software Product Line for the Bioinformatics Domain”, JBI, v. 56, p. 239-264.

Jansen, S., Finkelstein, A., Brinkkemper, S. (2009) “A Sense of Community: A Research Agenda for Software Ecosystems”, ICSE’09, p.187-190.

Mattoso, M., Werner, C., Travassos, G. H., Braganholo, V. (2010) “Towards supporting the life cycle of large scale scientific experiments”, IJBPIM, v. 5, p. 79-92,.

Roure, D., Goble, C., Stevens, R. (2009) “The Design and Realization of the myExperiment Virtual Research Environment for Social Sharing of Workflow”, Future Generation Computer Systems, p. 561–567.

Zhang, J. (2010) “Co-Taverna: A tool supporting collaborative scientific workflows”, IEEE 7th International Conference on Services Computing, SCC 2010, p. 41–48.

Zhang, J., Kuc, D., Lu, S. (2014) “Confucius: A tool supporting collaborative scientific workflow composition”, IEEE Trans. on Services Computing, 7(1), p. 2–17.