International Journal of Web Portals
Volume 9 • Issue 2 • July-December 2017

An Approach to Configuration Management of Scientific Workflows

Tassio Ferenzini Martins Sirqueira, Federal University of Juiz de Fora, Juiz de Fora, Brazil & Vianna Junior Institute, Juiz de Fora, Brazil

Regina Braga, Federal University of Juiz de Fora, Juiz de Fora, Brazil

Marco Antônio P. Araújo, Federal University of Juiz de Fora, Juiz de Fora, Brazil & Federal Institute of Southeast Minas Gerais, Juiz de Fora, Brazil

José Maria N. David, Federal University of Juiz de Fora, Juiz de Fora, Brazil

Fernanda Campos, Federal University of Juiz de Fora, Juiz de Fora, Brazil

Victor Stroële, Federal University of Juiz de Fora, Juiz de Fora, Brazil

ABSTRACT

A scientific software ecosystem aims to integrate all stages of an experiment and its related workflows, in order to solve complex problems. In this vein, in order to assure the experiment proper execution, any modification that occurs must be propagated to the associated workflows, which must be maintained and evolved for the successful conduction of the research. One way to ensure this control is through configuration management using data provenance. In this work, the authors use data provenance concepts and models, together with ontologies to provide an architecture for the storage and query of scientific experiment information. Considering the architecture, a proof of concept was conducted using workflows extracted from the myExperiment repository. The results are presented along the paper.

KEYWORDS

Data Provenance, Ontology, Scientific Experiment Management, Workflow Maintenance and Evolution

1. INTRODUCTION

A scientific experiment is defined as a series of interconnected operations (Goble *et al.*, 2010), which can be executed using one or more workflows. A scientific workflow is a model or template that represents a sequence of scientific activities implemented by tools in order to reach a certain objective (Deelman et al., 2009). The wide adoption of scientific workflows, as a mechanism to aggregate existing services, has radically revolutionized the way scientists conduct their experiments, since workflows allow to gather evidence for or against a hypothesis, and still demonstrate a known fact (Belhajjame et al, 2011).

According to (Nardi, 2009), users of scientific workflows, most of the time, work in a specific field of research and do not always have a computer science adequate training. Often, they begin an application by copying an existing workflow and then adjusting it to their needs. In this vein, another important issue is the loss of the researcher's knowledge about the experiment (Marinho et al., 2012), due to the delegation of tasks to computers that usually perform isolated actions, without documentation. Thus, to represent and support the development of a scientific experiment, it is necessary to register the associated workflows and their variations, since they can be modified during the research (Mattoso et al., 2010).

DOI: 10.4018/IJWP.2017070102

One way of storing this data is to use provenance models (Buneman et al., 2001), storing data produced from scientific workflows (Sirqueira et al., 2016). The use of provenance data allows the scientist to compose new workflows based on the reuse of data from previous ones. However, only provenance data used in isolation does not allow adequate control of the experiment and its associated workflows, making it difficult to manage the experiment as a whole. According to Hasan et al. (2007), it is necessary to use independent tools to manage the experiment and analyze its data, considering that Scientific Workflow Management Systems (SWMS) do not have this functionality. It considers only the researcher responsible for the workflow (Pereira et al., 2009), providing no collaboration mechanism, distribution and reuse support. This additional data, i.e., workflow versions, associated workflows, related experiments, and results are important for the publication of the experiment.

In this context, the objective of this work is to treat configuration management of scientific workflows throughout the experiment life cycle, based on the maintenance, evolution, and reuse of experiment´s data to improve the experimentation process and its use in other related contexts. Since each phase of the scientific experiment cycle presents specific tasks, and each modification on the execution of a task generates new versions of the workflow (Sirqueira et al., 2016), we consider this control essential for the proper execution and control of a scientific experiment. This article details the E-SECO ProVersion approach, which extends the E-SECO ecosystem (Freitas et al., 2015), to control and manage scientific workflows related to a given experiment, using provenance data and ontologies. In this vein, the research question can be defined as: Is E-SECO ProVersion architecture capable to derive maintenance and evolution information from experiments and related workflows?

Considering Figure 1, which details the experimentation life cycle of the E-SECO ProVersion approach, the configuration management is performed by the module "Configuration Management", which encompass the whole process.

Considering other similar works, such as myExperiment (Goble et al., 2010), CrowDLabs (Callahan, 2006) and SimiFlows (Silva et al., 2010), the differential of E-SECO ProVersion approach, are on 1) the research access to workflows repositories, 2) storage of the data consumed and generated by the experiment and workflows and 3) the data analysis functionalities, considering configuration management techniques. All the information is captured through web services and stored in a repository, allowing the query of provenance data, and analyzes using ontologies and inference engines.

This article has 7 sections besides this introduction. Section 2 presents the background of this research. Section 3 details the related works. In section 4, an analysis of scientific workflows repositories is presented. In Section 5, the E-SECO ProVersion platform is presented, detailing its architecture. Section 6 provides an evaluation of the platform and, in section 7, an analysis of the results. Finally, section 8 presents the conclusions.

2. BACKGROUND

Software maintenance activity is characterized by the revision of a delivered software product, for error correction, performance improvement or product adaptation to a new environment (Sommerville, 2003). In this context, Lehman (1996) presents the "Software Evolution Laws", which addresses eight aspects, usually observed in a software life cycle. It is not the intention of this work to discuss the evolution laws, however, some aspects can be applied in the experimentation maintenance and evolution context, such as: 1) the "Law of Continuous Change", which establishes that all software must be modified continuously or it will become less useful. Applying to the workflow context, as the research progresses, workflows tend to require evolution to remain useful; 2) The "Law of Increasing Complexity", which defines that with the evolution process, the structure tends to become more

Related Content