LODFlow – a Workflow Management System for Linked Data Processing

Sandro Rautenberg
Midwestern State University (UNICENTRO)
Guarapuava, PR, Brazil
srautenberg@unicentro.br

Ivan Ermilov
AKSW/BIS
University of Leipzig
Leipzig, Germany
iermilov@informatik.uni-leipzig.de

Edgard Marx
AKSW/BIS
University of Leipzig
Leipzig, Germany
marx@informatik.uni-leipzig.de

Sören Auer
University of Bonn / Fraunhofer IAIS, Germany
auer@cs.uni-bonn.de

Axel-Cyrille N. Ngomo
AKSW/BIS
University of Leipzig
Leipzig, Germany
ngonga@informatik.uni-leipzig.de

ABSTRACT

The extraction and maintenance of Linked Data datasets is a cumbersome, time-consuming and resource-intensive activity. The cost for producing Linked Data can be reduced by a workflow management system, which describes plans to systematically support the lifecycle of RDF datasets. We present the LODFlow Linked Data Workflow Management System, which provides an environment for planning, executing, reusing, and documenting Linked Data workflows. The LODFlow approach is based on a comprehensive knowledge model for describing the workflows and a workflow execution engine supporting systematic workflow execution, reporting, and exception handling. The environment was evaluated in a large-scale real-world use case. As result, LODFlow supports Linked Data engineers to systematically plan, execute and assess Linked Data production and maintenance workflows, thus improving efficiency, ease-of-use, reproducibility, reuseability and provenance.

The environment was evaluated in a large-scale real-world use case. As result, LODFlow supports Linked Data engineers to systematically plan, execute and assess Linked Data production and maintenance workflows, thus improving efficiency, ease-of-use, reproducibility, reuseability and provenance.

Categories and Subject Descriptors

H.4.1 [**Office Automation**]: Workflow management

General Terms

Linked Data, Workflow Management, Linked Data Workflow Management System.

SEMANTiCS '15, September 15-17, 2015, Vienna, Austria

ACM 978-1-4503-3462-4/15/09 ...$15.00.
DOI: http://dx.doi.org/10.1145/2814864.2814882

1. INTRODUCTION

The creation and maintenance of Linked Data datasets is currently performed by extensive manual workflows or proprietary scripts, which are not portable, reusable, and difficult to maintain. For Linked Data extraction and publication projects such as LinkedGeoData or DBpedia, a long number of activities is required in order to produce a particular version of the datasets. These activities include, for example, downloading and loading dumps of the raw data, running various extraction tools, performing quality analysis and sanity checks, loading the resulting data into a SPARQL endpoint for publication, generating links to other datasets etc. Performing such Linked Data production and maintenance activities requires in-depth knowledge of scripting languages, various RDF serialization and publication technologies, tools such as SILK, LIMES for linking, and standards such as R2RML for relational data mapping. As a result, such processes are cumbersome, time-consuming, and error-prone, requiring substantial skills.

In other domains, planning, executing, and orchestrating activities is supported by workflow management techniques and systems. With the *Business Process Execution Language* (BPEL), for example, a comprehensive language for the planning, execution, and orchestration of Web Services exists. There is also a large number of products from vendors including SAP, IBM, and Apache supporting BPEL available on the market. In addition, for scientific workflows, a number of languages and tools supporting the workflow design and execution exist. Specialized scientific workflow systems, e.g. *Discovery Net* [3], *Apache Taverna* [9] and *Kepler* [12], provide a visual programming front-end enabling user to construct their applications as a visual graph by connecting nodes together, and tools have been developed to execute such workflows in a platform-independent manner [10].

In this article, we present a comprehensive approach for defining, planning, orchestrating, and executing Linked Data production and maintenance workflows, the *Linked Open Data workFlow system* (LODFlow). LODFlow is based on

defining workflows comprising reusable activities and steps using a comprehensive ontological model, the *Linked Data Workflow Project Ontology* (LDWPO). We implemented a workflow execution environment, which takes workflows defined in LDWPO ontology as an input, executes the various steps defined in the ontology, and generates detailed reports about the execution and potential errors in machine and human-readable ways. The LODFlow is able to launch a number of specialized tools such as DBpedia DIEF or SparqlMap for extraction and mapping, SILK for linking, Luzzu for quality analysis. The LODFlow preserves provenance and adds comprehensive metadata, such as the version, invocation, and configuration of the tool execution in a concrete workflow instantiation.

The benefits of the LODFlow include:

1. **Explicitness** - the declarative description of Linked Data production and maintenance workflows makes the activities and processes explicit and easy to understand for Linked Data engineers in a high level of abstraction.
2. **Reusability** - the fine-grained definition of activities and workflow steps facilitates the reuse across workflows.
3. **Repeatability** - workflows can be easily executed over and over again, thus making them repeatable, facilitating testing and reliability.
4. **Efficiency** - the automatic workflow execution greatly reduces the required manual human intervention and thus improves efficiency.
5. **Ease of use** - the high-level declarative description of workflows and the generation of comprehensive execution reports simplifies the definition, execution and analysis of Linked Data production and maintenance workflows.

As a result, we hope that LODFlow and LDWPO will improve the creation and maintenance of Linked Data.

The rest of the paper is structured as follows. We discuss *Linked Data Workflow Management* and its requirements in Section 2. In Section 3 we present the LODFlow. The evaluation of LODFlow is discussed in Section 4, based on a large-scale, real-world use case. In Section 5 we outline related work. Finally, in Section 6 we conclude the paper and outline directions for the future work.

2. LINKED DATA WORKFLOW MANAGEMENT

The creation and maintenance of Linked Data datasets require substantial efforts and resources. These efforts should be planned, follow best practices, and be performed in a systematic way. The concept of Linked Data Workflow Management (LDWFM) described in this section provides a formal grounding for the systematic rendering of Linked Data management processes.

The core concept of LDWFM is a *Workflow*. The *Workflow* concept originated in the business management domain, where it is defined by *Workflow Management Coalition* as "the automation of a business process, in whole or part, during which documents, information or tasks are passed from one participant to another for action, according to a set of procedural rules" [18]. Essentially, a *Workflow* orchestrates steps that transform resources in order to reach a desired result. Benefits of systematic workflow management include explicit description of activities, reproducibility, automatization and generally increased efficiency.

According to [7], *Workflow Management* is a technology supporting the reengineering of business and information processes. The main components of the *Workflow Management* are a workflow definition and technological infrastructure. The workflow definition describes the aspects for coordination, processing and controlling of the resource lifecycle. The technological infrastructure provides means for efficient design and implementation of workflows. To facilitate *Workflow Management*, the technological infrastructure should fulfill the following requirements:

- be component-oriented, thus enabling integration and interoperability among loosely-coupled system and service components;
- support the implementation of a workflow, facilitating resource provenance and reproducibility;
- ensure the correctness and reliability of applications in the presence of concurrency and failures;
- support the evolution, replacement, and addition of the workflow steps in a reengineering process.

In a Linked Data environment, the *Workflow Management* should cover the specification, execution, registration, and control of procedures for producing and maintaining Linked Data datasets. Therefore, in a Linked Data context, we can define the requirements for LDWFM as follows.

REQUIREMENT 1. ***Linked Data Workflow Planning*** – *the capability for describing Linked Data datasets production strategy. Achieved by specifying a list of steps, where each step corresponds to a tool, tool configuration, as well as input and output datasets.*

REQUIREMENT 2. ***Linked Data Workflow Execution*** – *the capability for automating workflows, which involves a controlled environment for a plan execution. Achieved by running tools with tool configurations over input datasets.*

REQUIREMENT 3. ***Linked Data Workflow Reusability*** – *the capability to reuse the whole or a part of existing workflows.*

REQUIREMENT 4. ***Linked Data Workflow Documentation*** – *the capability to represent Linked Data Workflow plans and executions in human-readable formats.*

REQUIREMENT 5. ***Linked Data Workflow Repeatability*** – *the capability for reproducing the same result over time.*

3. LODFLOW - THE LINKED DATA WORKFLOW MANAGEMENT SYSTEM

In this section, we describe our *Linked Data Workflow Management System* (LDWMS), dubbed LODFlow. We understand an LDWMS as an extension of *Workflow Management Systems* (WMS) in the business management domain. A WMS "defines, creates and manages the execution of workflows through the use of software, running on one or more workflow engines, which is able to interpret the process, interact with workflow participants and, where required, invoke the use of IT tools and applications" [18]. Extending this definition to the Linked Data context and taking the requirements described in Section 2 into consideration, we define LODFlow as a system, consisting of the following components:

1. **Linked Data Workflow Knowledge Model** defines a common vocabulary for modeling, analyzing, executing, and documenting the Linked Data Workflows.
2. **Knowledge Base** records the data about Linked Data Workflows, according to the Linked Data Workflow Knowledge Model.
3. **Linked Data Workflow Maintenance Component** is a subsystem used for registering the Linked Data Plans and Linked Data Executions in the knowledge base.
4. **Linked Data Workflow Execution Engine** is a subsystem used for retrieving a planned workflow from the knowledge base and executing the plan in order to produce or maintain the datasets.
5. **Linked Data Workflow Report Component** is a subsystem used for generating report documents for Linked Data Workflows.

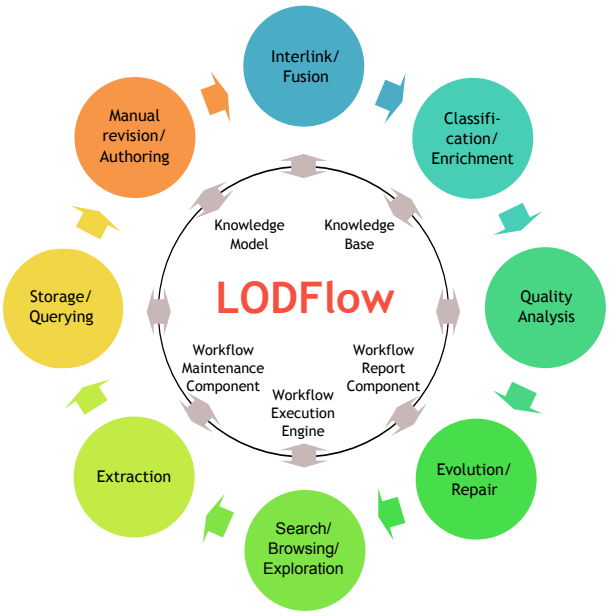

Figure 1: Linked Data Lifecycle supported by LODFlow.

LODFlow is designed in compliance with the Linked Data Lifecycle [2]. Therefore, it supports the orchestration of Linked Data tools for various aspects of the lifecycle [16].

In the following, we discuss LODFlow components in more detail.

3.1 Linked Data Workflow Knowledge Model and Knowledge Base

The *Linked Data Workflow Knowledge Model* is stored in the knowledge base adhering to the *Linked Data Workflow Project Ontology* (LDWPO)[1], an ontology that standardizes the method, plan, and execution concepts for operationalizing the production of Linked Data datasets.

The LDWPO[2] is developed following the best practices, in particular by using approaches proposed by the *On-to-Knowledge* [15], the *METHONTOLOGY* [8], and the *Ontology Development 101 Guide* [14]. We extend the *Publishing Workflow Ontology* (PWO) [6], the *Open Provenance Model Vocabulary* (OPMV) [13], and the *PROV Ontology* (PROV-O) [11] (depicted in Figure 2) in such a way, which allows: **i)** to model the creation of Linked Data datasets in the context of Linked Data Lifecycle; **ii)** to plan workflows for Linked Data datasets maintenance, thus enabling provenance extraction and reproducibility over time; and **iii)** to execute workflows in a (semi-)automatized way using Linked Data Stack technologies.

The main concept of LDWPO is `LDWProject`, which represents the endeavour for creating or maintaining `RDFDatasets`. In addition to annotation and metadata properties, an `LDWProject` is associated with one or several `LDWorkflows`. An `LDWorkflow` embodies the plan necessary to produce (upper part of Figure 2) `RDF Datasets`, encapsulating a sequence of `LDWSteps`. `LDWStep` is a concept that represents an atomic and reusable unit of an `LDWorkflow`. It describes the processing of a set of input `Datasets`, using a `Tool` in a specified version with a concrete `Tool Configuration`, in order to produce a set of output `Datasets`. An `LDWStep` can be: **i)** reused by different `LDWorkflows` within existing `LDWProjects`; and **ii)** executed automatically in a controlled environment on a user request. Those features are discussed in more detail in Section 4, where we present a real-world use case.

Once an `LDWorkflow` is formulated, it can be reused as a `Plan` in an `Execution` at any particular point of time. In LDWPO, the concept describing an instantiation for executing an `LDWorkflow` is `LDWorkflowExecution`. `LDWorkflowExecution` aggregates the sequence of `LDWStepExecutions` that corresponds to the sequence of `LDWSteps` of a particular `LDWorkflow`. During an `LDWorkflowExecution` run, the corresponding `LDWStepExecutions` can generate `Messages` such as D2R logging report and `Statuses` such as successfully finished, unsuccessfully finished, aborted, etc. `LDWorkflowExecution` holds the information about a particular run of an `LDWorkflow` at a certain point in time, thus enabling provenance tracing and later verification (bottom part of Figure 2).

[1] `http://aksw.org/Projects/LDWPO.html`
[2] A detailed technical report for LDWPO is available at: `https://github.com/AKSW/ldwpo/blob/master/misc/technicalReport/LDWPO_technicalReport.pdf`

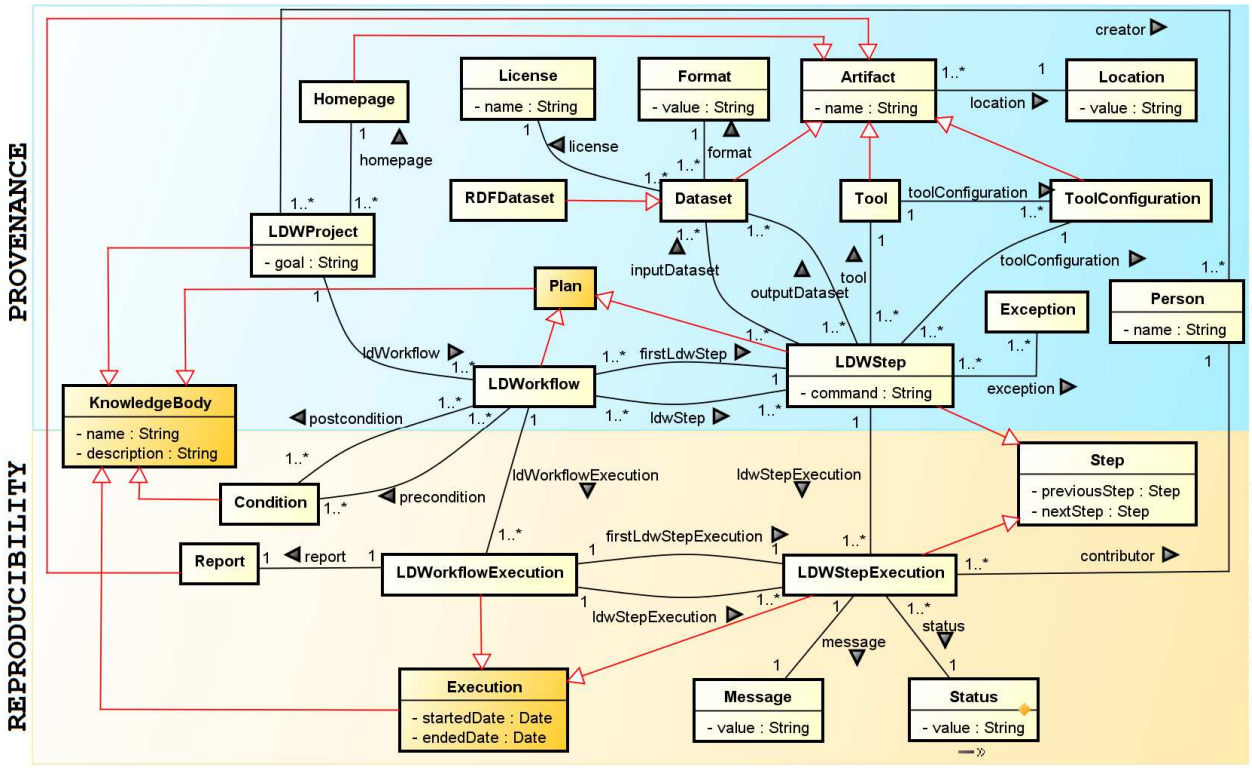

Figure 2: LDWPO ontology overview for describing and planning LODFlow workflows.

3.2 Linked Data Workflow Maintenance Component

As the *Linked Data Workflow Maintenance Component*, we adopted the *NeOn Toolkit*[3]. It is an open source multi-platform ontology engineering environment, which provides comprehensive support for the ontology engineering lifecycle [4]. NeOn facilitates the human-ontology interaction, which is useful for registering `LDWProject`s, `LDWorkflow`s, `LDWStep`s, `LDWorkflowExecution`s, `LDWStepExecution`s in the knowledge base.

3.3 Linked Data Workflow Execution Engine

The *Linked Data Workflow Execution Engine* is the main component for producing Linked Data datasets. It is responsible for retrieving the `LDWorkflow` from the knowledge base, interpreting the `LDWorkflow`s according to an established *Knowledge Model*, and managing the pipeline for producing the Linked Data datasets. For performing these tasks and interacting with the other components, we developed a tool for interpreting the resources from LDWPO and invoking other `Tool`s. This tool is dubbed *LODFlow Engine*[4].

3.4 Linked Data Workflow Report Component

The *Linked Data Workflow Report Component* is a component for generating technical reports for `LDWProject`s, `LDWorkflow`s, and `LDWorkflowExecution`s. We implemented this component with the *LODFlow Report Tool*[5]. In its current version, LODFlow Report is able to generate reports for a whole `LDWProject`.

4. LODFLOW IN USE

In this section, we describe a comprehensive LODFlow use case for supporting the production and maintenance of a *5 star RDF dataset*[6]. The dataset is derived from the *Qualis Index*[7], a Brazilian scientometric scoring database for journals world-wide.

Qualis is created and used by the *Brazilian Research Community*. A typical entry in *Qualis* consists of ISSN, journal name, related knowledge field, and qualified journal score. As main uses, *Qualis* data is adopted in bibliometric and scientometric assessments, as well as for ranking post-graduate programs, research proposals, or individual research scholarships.

Although a web interface is publicly available for querying *Qualis* data, it has several limitations: **i)** the data is available only as `1 Star Data`, i.e. in PDF format; **ii)** there is no version control, i.e. only the current version of the dataset is available; and **iii)** the data is not linked to other datasets, which limits its usefulness.

[3] `http://neon-toolkit.org/wiki/Main_Page.html`
[4] `https://github.com/AKSW/LODFlow/tree/master/tools/LODFlowEngine`
[5] `https://github.com/AKSW/LODFlow/tree/master/tools/LODFlowReport`
[6] For more information, please see the data classification system proposed by Tim Berners-Lee at `http://5stardata.info/`
[7] `http://qualis.capes.gov.br/webqualis/principal.seam`

In order to overcome these limitations, we extracted the data of the last nine years from the web interface. The data is converted to a `5-star` dataset and interlinked to the *DBpedia* knowledge base. These efforts are formally described in the *QualisBrasil* `LDWProject` (Figure 3), which enables to utilize a single `LDWorkflow` to convert *Qualis* data to the 5-star RDF dataset. The workflow can be executed repeatedly, for example, when there is a new version of the dataset published each year. In the following, we present how the *QualisBrasil* `LDWProject` is planned and executed using LODFlow.

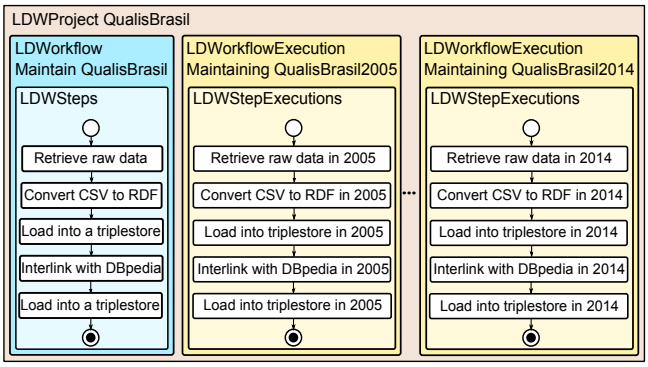

Figure 3: Utilizing a single workflow to convert Qualis Index data in 2005, ..., and 2014 (LODFlow).

4.1 Linked Data Workflow Planning

In order to verify the LODFlow adherence to the Requirement 1 – *Linked Data Workflow Planning*, we utilize the NeOn Toolkit and the LDWPO[8]. These components constitute the Linked Data Workflow Maintenance Component and the Linked Data Workflow Knowledge Model, accordingly. Using the NeOn Toolkit and LDWPO, we instantiate the *QualisBrasil* `LDWProject` and the *Maintain QualisBrasil* `LDWorkflow` and store them in the knowledge base. Additionally, for the `LDWorkflow`, we create five `LDWStep`s as follows:

1. ***LDWStep a*** includes the tasks for retrieving data from a legacy database and saving it in a CSV format;
2. ***LDWStep b*** performs the task of converting the CSV data to the *Qualis* `RDFDataset` using the RDB2RDF transformation tool *Sparqlify*[9];
3. ***LDWStep c*** loads the generated `RDFDataset` into a triplestore;
4. ***LDWStep d*** interlinks the resulting *Qualis* `RDFDataset` with *DBpedia* data using the link discovery tool *LIMES*[10]; and
5. ***LDWStep e*** loads the acquired links into a triplestore.

In the listing below, we show an excerpt from the whole workflow specification: the `LDWorkflow` description and `LDWStep b` as an illustrating example. The `LDWStep` is instantiated with the following properties: name (line 2); step description (line 3); a `command`, which defines a specific script or operation for an automatic step processing (line 4); the input and output `Dataset`s (lines 5 and 6); the `Tool` with its `Tool Configuration` (lines 7 and 8); and the `exception`, a property used for registering actions to be taken upon an exception (line 9). Thus the provenance information (e.g. about the particular dataset version and tool configuration) is registered in the knowledge base adhering to the *Knowledge Model*. In order to define the production strategy for a Linked Data dataset, each `LDWStep` is declared to be a part of an `LDWorkflow` (lines 27 to 31). The production strategy can be reused for producing the Linked Data datasets by running the `LDWStepExecution`s (lines 13 to 15). Therefore, the LODFlow fulfills the Requirement 1.

```
1  :ldwStep_02_qualisBrasil
2      :name "LDWStep b"^^xsd:string ;
3      :description "convert the qualis data from CSV
           format to NT format by using the Sparqlify
           "^^xsd:string ;
4      :command "real/QualisBrasilProject/bin/
           applyingSparqlify.sh"^^xsd:string ;
5      :inputDataset :dataset_evaluations_csv ;
6      :outputDataset :dataset_evaluations_nt ;
7      :tool :tool_sparqlify ;
8      :toolConfiguration :
           toolConfiguration_ldwStep_02_qualisBrasil ;
9      :exception :abort ;
10     :previousStep :ldwStep_01_qualisBrasil ;
11     :nextStep :ldwStep_03_qualisBrasil ;
12     :ldwStepExecution
13         :ldwStepExecution_02_qualisBrasil2005 ,
14         [...] ,
15         :ldwStepExecution_02_qualisBrasil2014 ;
16     a :LDWStep, owl:NamedIndividual .
17
18 :ldWorkflow_maintaining_qualisBrasil
19     :name "Maintain QualisBrasil"^^xsd:string ;
20     :description "Workflow applied to create Linked
           Data dataset of Qualis Periodicals scores (
           all years), in a automatized way."^^xsd:
           string ;
21     :firstLdwStep :ldwStep_01_qualisBrasil ;
22     :ldWorkflowExecution
23         :ldWorkflowExecution_for_qualisBrasil2005 ,
24         [...]
25         :ldWorkflowExecution_for_qualisBrasil2014 ;
26     :ldwStep
27         :ldwStep_01_qualisBrasil ,
28         :ldwStep_02_qualisBrasil ,
29         :ldwStep_03_qualisBrasil ,
30         :ldwStep_04_qualisBrasil ,
31         :ldwStep_05_qualisBrasil ;
32     [...]
33     a :LDWorkflow, owl:NamedIndividual .
```

4.2 Linked Data Workflow Execution

To fulfill the Requirement 2 – *Linked Data Workflow Execution*, a planned `LDWorkflow` and its `LDWStep`s should be executed one or multiple times. In LODFlow, the *LODFlow Engine* is responsible for performing this operation.

In the following listing, we show that the *LODFlow Engine* can be used through its command-line interface to retrieve all information required for an `LDWorkflowExecution` (such as dataset locations, tool locations, parameters, and configurations). Therefore, the *LODFlow Engine* enables the processing of Linked Data datasets in an automatized way. Additionally, for each `LDWStepExecution`, the engine aggre-

[8]LDWPO and data related to *QualisBrasil* `LDWProject` is available at: `https://github.com/AKSW/LODFlow/blob/master/useCases/QualisBrasil/ontology/ldwpo.owl`
[9]`http://aksw.org/Projects/Sparqlify.html`
[10]`http://aksw.org/Projects/LIMES.html`

gates the log and error messages of the tools, which can be visualized later as a management `Report`. By using the engine, LODFlow deals with the automation aspects of workflow executions, thus fulfilling the Requirement 2.

```
# for executing lodflowEngine uses:
java -jar lodflowEngine.jar "ontology filename" "
     LDWProject name" "LDWorkflowExecution name"

# running a workflowExecution for QualisBrasil 2014.
java -jar tools/lodflowEngine.jar "ontology/ldwpo.owl
     " "ldwpo:project_QualisBrasil" "ldwpo:
     ldWorkflowExecution_maintaining_qualisBrasil_2014
     " > reports/managementReport_QualisBrasil_2014.
     html
```

4.3 Linked Data Workflow Reusability

The Requirement 3 – *Linked Data Workflow Reusability* is fulfilled by the system if the whole or a part of existing `LDWorkflow`s can be reused. To demonstrate LODFlow compliance with this requirement, we reused the *Maintain QualisBrasil* `LDWorkflow` over ten `LDWorkflowExecution`s. This resulted in ten corresponding *QualisBrasil* `RDFDatasets`, from 2005 to 2014.

In the Table 1, we represent annual *LDWorkflowExecutions* from 2005 to 2014 for the production of the *QualisBrasil RDFDataset*. We show the amount of processed data as well as processing time for each `LDWStep`. The experiment was carried out running Ubuntu 12.04.5 LTS inside a VirtualBox virtual machine with 4GB of RAM and one CPU core. The host machine was Intel Core i7-4702MQ processor, 64GB SSD, 8GB RAM. The experiment shows, that LODFlow fulfills the Requirement 3.

4.4 Linked Data Workflow Documentation

As result of an *LDWorkflowExecution*, LODFlow generates human-readable reports, which are useful for managing the `LDWProject`s. LODFlow is able to generate reports for the whole project.

To show that LODFlow fulfills the Requirement 4 – *Linked Data Workflow Documentation*, we use the *LODFlow Report tool* to generate a complete report[11] for the *QualisBrasil* `LDWProject`. Such reports are essential for collecting statistics. For instance, Table 1 and Table 2 are compiled using the data from the generated report.

4.5 Linked Data Workflow Repeatability

To fulfill the Requirement 5 – *Linked Data Workflow Repeatability*, LODFlow should be able to reproduce same results over time. Using *LODFlow Engine*, we performed each annual `LDWorkflowExecution` five times. In the Table 2, we summarize the average processing time, number of triples for each annual `LDWorkflowExecution`, and previous results (from Table 1). By analyzing the Table 2, we observe that the reproduction of *QualisBrasil* `RDFDatasets` using LODFlow is guaranteed. Specifically, we can see that the number of triples remains the same for all `LDWorkflowExecution` runs, while the processing time standard deviation is small.

[11] `https://github.com/AKSW/LODFlow/blob/master/useCases/QualisBrasil/reports/reportQualisBrasil.html`

	Average & standard deviation		previous results	
Year	# saved triples	processing time (minutes)	# saved triples	processing time (minutes)
2005	216,225 ± 0.0	1:22 ± 0:09	216,225	1:16
2006	391,327 ± 0.0	1:59 ± 0:02	391,327	2:02
2007	566,429 ± 0.0	2:37 ± 0:08	566,429	2:32
2008	852,840 ± 0.0	3:54 ± 0:14	852,840	3:54
2009	1,124,007 ± 0.0	5:09 ± 0:07	1,124,007	5:13
2010	1,395,174 ± 0.0	6:29 ± 0:20	1,395,174	6:47
2011	1,968,561 ± 0.0	9:01 ± 0:23	1,968,561	9:13
2012	2.505,711 ± 0.0	11:13 ± 0:18	2.505,711	12:00
2013	3,042,858 ± 0.0	14:30 ± 0:30	3,042,858	14:32
2014	3,590,448 ± 0.0	16:01 ± 0:23	3,590,448	15:38

Table 2: Comparing the reproducibility of Maintaining QualisBrasil LDWorkflowExecutions.

5. RELATED WORK

To the best of our knowledge, this is the first work that specifically addresses Workflow Management for Linked Data.

Workflow, *Workflow Management*, and *Workflow Management System* are concepts long established in the business management domain. For promoting business process automation, enterprises use *Enterprise Resource Planning* (ERP) to manage the workflow executions. To provide business solutions, important players such as SAP, IBM, and Oracle, among others adopt *Business Process Execution Language* (BPEL) [17] in their technologies as the language to describe web services for workflow executions. In the Semantic Web area, there were quite a number of approaches (e.g. OWL-S, SA-WSDL) developed to enable semantic descriptions of Web Services and Service Oriented Architectures thus facilitating business process and workflow planning and execution [5].

In another domain, the scientific community coined the term *Scientific Workflow* as "the automated process that combines data and processes in a structured set of steps to implement computational solutions to a scientific problem" [1]. To facilitate workflows for data and control sequences, *Scientific Workflow Management Systems* such as *Apache Taverna* [9] and *Kepler* [12] were developed. These management systems employ ontologies for modeling the workflows, such as Scufl2 and Kepler ontologies, respectively. At the time of writing, the *Scufl2 ontology* is not available[12]. *Kepler* ontologies are the part of the Kepler framework and can be found in the source code. *Kepler* ontologies do not include human-readable descriptions for concepts, as we show in the following listing. Concept descriptions are required to facilitate the reuse of ontology resources. In our vision, the absence of such descriptions limits the adoption of *Kepler* ontologies. To leverage the limitations of Scufl2 and Kepler ontologies, we designed LDWPO to support LODFlow.

[12] the ontology is not published `http://taverna.incubator.apache.org/documentation/scufl2/ontology` 08-06-2015 10:00

LDWorkflow-Execution (year)	LDWStep a # journal scores	LDWStep a step time (%)	LDWStep b # generated triples	LDWStep b step time (%)	LDWStep c # saved triples	LDWStep c step time (%)	LDWStep d # interlinked journals	LDWStep d step time (%)	LDWStep e # saved triples	LDWStep e step time (%)	processing time (minutes)
2005	35,020	1.15	525,315	17.35	214,157	22.95	2,068	35.37	216,225	23.18	1:16
2006	70,040	1.30	1,050,615	16.00	389,259	30.63	2,068	23.55	391,327	28.52	2:02
2007	105,060	1.49	1,575,915	15.96	564,361	32.49	2,068	17.50	566,429	32.56	2:32
2008	159,293	1.49	2,389,410	15.35	850,096	33.92	2,744	14.76	852,840	34.48	3:54
2009	213,526	1.47	3,202,905	14.15	1,121,263	35.96	2,744	11.40	1,124,007	37.02	5:13
2010	267,759	1.49	4,016,400	14.45	1,392,430	37.80	2,744	8.66	1,395,174	37.60	6:47
2011	375,188	1.59	5,627,835	13.65	1,965,025	37.87	3,536	9.00	1,968,561	37.89	9:13
2012	482,617	1.46	7,239,270	13.00	2,502,175	41.09	3,536	6.92	2,505,711	37.53	12:00
2013	590,046	1.68	8,850,705	13.85	3,039,322	39.54	3,536	5.92	3,042,858	39.01	14:32
2014	698,668	2.14	10,480,035	14.25	3,586,412	37.66	4,036	5.75	3,590,448	40.20	15:38

Table 1: Producing QualisBrasil RDFDataset. Annual LDWorkflowExecutions from 2005 to 2014.

```
[...]
  <owl:Class rdf:ID="Workflow">
    <rdfs:label rdf:datatype="http://www.w3.org/2001/
        XMLSchema#string">Workflow</rdfs:label>
  </owl:Class>

  [...]

  <owl:Class rdf:ID="WorkflowOutput">
    <rdfs:label rdf:datatype="http://www.w3.org/2001/
        XMLSchema#string">Workflow Output</rdfs:
        label>
    <rdfs:subClassOf>
      <owl:Class rdf:about="#DataOutput"/>
    </rdfs:subClassOf>
  </owl:Class>
[...]
```

6. CONCLUSION AND FUTURE WORK

In this paper, we presented LODFlow, an LDWMS for supporting the Linked Data dataset production. In our vision, an established LDWMS should cover the specification, execution, registration, and control of procedures for reproducing Linked Data datasets over time. In this way, we designed such system, based on five components: a *Linked Data Workflow Knowledge Model*, a knowledge base, a *Linked Data Workflow Maintenance Component*, a *Linked Data Workflow Execution Engine*, and a *Linked Data Workflow Report Component*. We see this work as a first step in a larger research and technology development agenda, which aims at providing comprehensive workflow support for Linked Data production and maintenance processes.

Noteworthy, LODFlow is already used in a real-world application for facilitating bibliometric and scientometric researches in Brazil. As the result, Qualis RDF dataset[13] is maintained in an automated fashion and publicly available at `http://lodkem.led.ufsc.br:8890/sparql`.

As shown in the use case, LODFlow can tackle one of the most pressing and challenging problems of Linked Data management – the automatization of processing workflows, which are currently cumbersome, resource demanding and inefficient. The benefits of *explicitness*, *reusability*, *repeatability*, *efficiency*, and *ease of use* are observed when LODFlow is applied. In particular, with LODFlow is possible to create comprehensive workflow descriptions, preserving provenance information for reproducing the `LDWorkflows` of an `LDWProject`. Moreover, technologically, it is possible to mediate the use of tools, enabling the automatized execution of `LDWorkflows` in the context of the Linked Data Stack and Linked Data Lifecycle.

As future work, we aim to improve LODFlow, as well as adopt it in further use cases. For instance, as NeOn Toolkit interface cannot be customized, in the next effort, we foresee the development of another *Linked Data Workflow Maintaining Component*, with an easy-to-use interface for Linked Data engineers. At the moment of writing LODFlow is limited to the processing of sequential workflows. We aim to improve its *Linked Data Knowledge Model* for supporting branching. Moreover, we plan to incorporate LODFlow into the Linked Data Stack, providing a full-integrated support for the `LDWProject` modeling and management.

[13]published on datahub at `http://datahub.io/dataset/qualisbrasil`

Acknowledgment

This work was supported by the Brazilian Federal Agency for the Support and Evaluation of Graduate Education (CAPES/Brazil), under the program Sciences without Borders (Process number - 18228/12-7). We acknowledge support from GeoKnow project, GA number no. 318159, as well as BMBF project SAKE.

7. REFERENCES

[1] Ilkay Altintas, Oscar Barney, and Efrat Jaeger-Frank. Provenance collection support in the kepler scientific workflow system. In Luc Moreau and Ian T. Foster, editors, *IPAW*, volume 4145 of *Lecture Notes in Computer Science*, pages 118–132. Springer, 2006.

[2] Sören Auer. Introduction to lod2. In Sören Auer, Volha Bryl, and Sebastian Tramp, editors, *Linked Open Data – Creating Knowledge Out of Interlinked Data*. Springer-Verlag, 2014.

[3] V. Ćurčin, M. Ghanem, Y. Guo, M. Köhler, A. Rowe, J. Syed, and P. Wendel. Discovery net: Towards a grid of knowledge discovery. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '02, pages 658–663, New York, NY, USA, 2002. ACM.

[4] Michael Erdmann and Walter Waterfeld. Overview of the neon toolkit. In María del Carmen Suárez-Figueroa, Asunción Gómez-Pérez, Enrico Motta, and Aldo Gangemi, editors, *Ontology Engineering in a Networked World*, pages 281–301. Springer, 2012.

[5] Dieter Fensel, Federico Michele Facca, Elena Simperl,

and Ioan Toma. *Semantic web services*. Springer Science & Business Media, 2011.

[6] Aldo Gangemi, Silvio Peroni, David Shotton, and Fabio Vitali. A pattern-based ontology for describing publishing workflows. In *Proceedings of the 5th Workshop on Ontology and Semantic Web Patterns (WOP2014) co-located with the 13th International Semantic Web Conference (ISWC 2014), Riva del Garda, Italy, October 19, 2014.*, pages 2–13, 2014.

[7] Dimitrios Georgakopoulos, Mark F. Hornick, and Amit P. Sheth. An overview of workflow management: From process modeling to workflow automation infrastructure. *Distributed and Parallel Databases*, 3(2):119–153, 1995.

[8] Asunción Gomez-Perez, Mariano Fernandez-Lopez, and Oscar Corcho. *Ontological Engineering: With Examples from the Areas of Knowledge Management, E-Commerce and the Semantic Web, 1st Edition*. Springer-Verlag, Heidelberg, 2004.

[9] D Hull, K Wolstencroft, R Stevens, C Goble, M R Pocock, P Li, and T Oinn. Taverna: a tool for building and running workflows of services. *Nucleic Acids Res*, 34(Web Server issue):729–732, July 2006.

[10] D. Johnson, K. Meacham, and H. Kornmayer. A middleware independent grid workflow builder for scientific applications. In *E-Science Workshops, 2009 5th IEEE International Conference on*, pages 86–91, Dec 2009.

[11] Timothy Lebo, Satya Sahoo, Deborah McGuinness, Khalid Belhajjame, James Cheney, David Corsar, Daniel Garijo, Stian Soiland-Reyes, Stephan Zednik, and Jun Zhao. PROV-O: The prov ontology. Retrieved from http://www.w3.org/TR/prov-o/ on 13.01.2015.

[12] Bertram Ludäscher, Ilkay Altintas, Chad Berkley, Dan Higgins, Efrat Jaeger, Matthew Jones, Edward A. Lee, Jing Tao, and Yang Zhao. Scientific workflow management and the kepler system. *Concurrency and Computation: Practice and Experience*, 18(10):1039–1065, 2006.

[13] Luc Moreau, Ben Clifford, Juliana Freire, Joe Futrelle, Yolanda Gil, Paul Groth, Natalia Kwasnikowska, Simon Miles, Paolo Missier, Jim Myers, Beth Plale, Yogesh Simmhan, Eric Stephan, and Jan Van den Bussche. The open provenance model core specification (v1.1). *Future Generation Computer Systems (FGCS)*, 27(6):743–756, 2011. [IF 1.978, CORE A].

[14] Natalya F Noy and Deborah L McGuinness. Ontology development 101: A guide to creating your first ontology. *Development*, 32(1):1–25, 2001.

[15] York Sure and Rudi Studer. On-To-Knowledge methodology. In John Davies, Dieter Fensel, and Frank van Harmelen, editors, *On-To-Knowledge: Semantic Web enabled Knowledge Management*, chapter 3, pages 33–46. J. Wiley and Sons, 2002.

[16] Bert Van Nuffelen, Valentina Janev, Michael Martin, Vuk Mijovic, and Sebastian Tramp. Supporting the linked data life cycle using an integrated tool stack. In Sören Auer, Volha Bryl, and Sebastian Tramp, editors, *Linked Open Data – Creating Knowledge Out of Interlinked Data*. Springer-Verlag, 2014.

[17] Sanjiva Weerawarana, Francisco Curbera, Frank Leymann, Tony Storey, and Donald F Ferguson. *Web services platform architecture: SOAP, WSDL, WS-policy, WS-addressing, WS-BPEL, WS-reliable messaging and more*. Prentice Hall PTR, 2005.

[18] WfMC. Wfmc: Terminology and glosssary. Online PDF, February 1999.