Basic information

Project name	LinkedPipes Applications
Abbreviation	LPA
Supervisor	Jiří Helmich <helmich@ksi.mff.cuni.cz>
Consultants	Jakub Klímek <klimek@ksi.mff.cuni.cz> Martin Nečaský <martinnec@gmail.com>
Annotation	The goal of the project is to create a new component of the LinkedPipes platform - LinkedPipes Applications. The tool should be focused on lay users and enable them to work with data published according to Linked Data principles. This new application will provide the user with a simple UI interface that will orchestrate other tools from the LinkedPipes platform on background, especially LinkedPipes ETL and LinkedPipes discovery.

Motivation

Open data are more and more often published in the form of Linked Data. This form is based on the RDF data model, which allows expressing not just the data itself, but also its semantics. The semantic description is implemented by reusing shared vocabularies, which are specified mainly by the W3C consortium as web standards, but anyone can create and register their own in an open catalogue like Linked Open Vocabularies (LOV). However, there is still a lack of tools that would allow lay users to easily consume (exploration, visualization, etc.) such data, especially without any technological knowledge of Linked Data, RDF and other related technologies, just by providing a simple UI, while understanding well-known vocabularies for specific data types.

Project description

The goal of the project is to create a tool that would enable a non-expert (lay) user to use Linked Data easily, without any technical knowledge. The tool will contain a set of visual applications (for different types of data) that will allow the user to browse through datasets. To enable this functionality to a wide spectrum of datasets, the applications will implement well-known vocabularies specialized on the respective data types. As there are duplicate vocabularies defining different shapes for the same type of data, the tool has to provide a solution for transforming the data between different vocabularies. As we require the user to have no technical knowledge, this transformation has to be fully automatic. The tool therefore has to implement a set of transformation components that will allow this. By using LinkedPipes Discovery, the tool has to be able to discover transformation pipelines and execute them via LinkedPipes ETL. The result of the transformation pipeline will be provided to the user in a visual form.

The tool should support two different groups of users. First of all, it should provide help to domain experts (e.g. data journalists), who understand the published data (domain specialists) on a logical level, but don’t know neither Linked Data principles or the RDF format. Such users nowadays

prefer to convert Linked Data (so-called 5-star data) into a lesser format (e.g. XLS, CSV, JSON, atp.) as the tools they are used to working with (starting with Python scripts, ending with advanced analytical tools like Tableau or PowerBI) enable them to consume data more easily. These users need applications that allow them to preview and browse data sources easily and comfortably. They also often prepare visualizations and interactive miniapplications for lay users (e.g. readers).

Most of the time, lay users are not domain experts. They also lack technological knowledge of Linked Data and RDF. That's why they need a middle man, who would prepare the data into an understandable (most of the time also simplified) form, such as an interactive visualization. Domain experts can achieve this not only by a narrower choice of initially displayed data, but also e.g. by restricting the possibilities the lay users have available while browsing the dataset (e.g. less options in the user interface, predefined and fixed values of filter controls, etc.).

A good example of such an interactive application could be a browser for OLAP cube data. An expert user is presented with a complete n-dimensional data cube based on default settings of the application that were chosen automatically so that the application is able to offer some kind of preview to the user, e.g. a bar chart. The user can, however, decide they want any different view on the data cube (e.g. they would be interested in different cells of the cube, they will apply some OLAP operations, e.g. roll-up of a chosen dimension, etc.). Also, they might want to be able to determine, which data should be displayed to their readers by default and restrict operations that the reader can perform (e.g. fix, which values of individual dimensions are chosen for data presentation and let the user switch only between individual measures of the data cube).

A domain expert requires the following features:

- sign in
- management of published applications and their configurations
- management of data sources
- creation of a new instance of an interactive application
- a basic mode of an interactive application
- a configuration mode of an interactive application, e.g.:
    - configuration of default view properties (default values of user controls)
    - configuration of values that are available for user selection within user controls
        - removing whole filters (ignoring values of certain data properties)
        - removing some values of some filters
        - setting a fixed values for some filters
- an interface for publishing configured interactive applications (incl. generating a link and code for embedding to a 3rd party website) with a possibility to regularly refresh the data from a previously chosen data source.

An end user requires the following features:

- ability to access and use a pre-configured interactive application
- ability to control the interactive application by its own user interface

A developer requires the following features:

- API for a simple creation of other types of interactive applications
- documentation of the API

What is very important to note is that the size of the data sources is not known in advance, which means that the application has to be able to work smoothly with any size of data and allow its user to browse all of it.

The implemented tool is expected to deliver the following:

a) automatic recommendation of suitable interactive applications for a user-given data source while exploiting the features of LinkedPipes Discovery and LinkedPipeps ETL tools
b) a framework for developers that will allow for creation of user applications (previews, visualizations with filtering, etc.) and its documentation
c) an implementation of specified user application for manipulation with different types of data chosen based on RDF vocabularies:
    i) map data (coordinates, polygons, heatmaps, quantified places, etc.)
    ii) time data (events, intervals)
    iii) hierarchies
    iv) OLAP
d) user applications will support the following modes:
    i) preview - an application will receive automatically prepared data and allow the user to work with them (e.g. visualize it, offer a preview)
    ii) configuration - an application will allow a domain specialist to specify a subset of the data that is relevant for the lay user, i.e. based on the preprocessed data, they will build a suitable user interface, e.g. filters that are configurable
    iii) view - this mode is focused on the lay user, who will be presented with an application that was pre-configured in the configuration mode by a domain specialist

The implemented tool is expected to use the following tools via their APIs to deliver the above functionality:

- LinkedPipes ETL is designed to transform data using so-called transformation pipelines that can be built either manually by a user or through an API. The implemented tool can use LP-ETL to transform data between different RDF vocabularies from proprietary vocabularies to standard ones or those that were chosen to be supported.
- LinkedPipes Discovery is designed to automatically discover the aforementioned transformation pipelines based on a given input (a list of data sources and list of applications and their RDF input data descriptions). The implemented tool can use LP Discovery to automatically assemble the transformation pipelines.

Platform, technologies

It is recommended to use JavaScript and React library to implement the user interface. The choice of the backend technologies is more open, but since the best RDF libraries are written in Java, it is

highly recommended to use them and therefore build the backend using some JVM-enabled language and technologies.

Difficulty estimate

The project will be carried out by 5 team members while following the agile methodology of development. The deadline is 9 months from the start of the work on the project.

Work plan:

1st month	● setting up team processes (VCS, e-mail group, issue tracking, etc.) ● getting familiar with Linked Data, RDF ● getting familiar with LinkedPipes platform ● requirement analysis ● determination of data types for which interactive application will be implemented for (map, data cube, …)
2nd month	● analysis and design of individual applications ● determination of common application parts and components that will be implemented into the framework itself ● design of the framework API (will be implemented by individual apps)
3rd month	● deployment via Docker
4th month	● framework implementation ● applications implementation
5th month	● testing of the current implementation ● requirements refinement
6th month	● revision of the framework implementation based on the refined requirements
7th month	● testing of the current implementation ● documentation
8th month	● debugging ● acceptance testing ● documentation finalization
9th month	

Vymezení projektu

Projekt je zaměřen na následující oblasti (zaškrtněte vyhovující):

Diskrétní modely a algoritmy	
	diskrétní matematika a algoritmy
	geometrie a matematické struktury v informatice
	optimalizace
Teoretická informatika	
	Teoretická informatika
Softwarové a datové inženýrství	
x	softwarové inženýrství
x	vývoj software
x	webové inženýrství
	databázové systémy
x	analýza a zpracování rozsáhlých dat
Softwarové systémy	
	systémové programování
	spolehlivé systémy
	výkonné systémy
Matematická lingvistika	
	počítačová a formální lingvistika
	statistické metody a strojové učení v počítačové lingvistice
Umělá inteligence	
	inteligentní agenti
	strojové učení
	robotika
Počítačová grafika a vývoj počítačových her	
	počítačová grafika
	vývoj počítačových her