# Workflow supporting toolset for diagram-based collaborative ontology development implemented in the open budget domain

Dmitry Mouromtsev[1], Dmitry Pavlov[1,2], Yury Emelyanov[1,2], Alexey Morozov[2], Daniil Razdyakonov[2], and Olga Parkhimovich[2]

[1] ITMO University, St.Petersburg, Russia
mouromtsev@mail.ifmo.ru, yuvemelyanov@corp.ifmo.ru,
olya.parkhimovich@gmail.com
[2] Vismart Ltd., St.Petersburg, Russia
dmitry.pavlov@vismart.biz, alexey.morozov@vismart.biz,
daniil.razdyakonov@vismart.biz

**Abstract.** We present a live demo of a use case and a technical solution that addresses the problem of organizing the collaborative ontology development with deliverables including the diagrams and various views of the data model. The use case describes the real life situation, in which the geographically distributed team was challenged with a task of producing the open budget ontology and consequently was to select the tool set to support such development. The technical solution is based on the combination of 3 basic tools: Protégé - to provide a collaborative environment for ontology creation and modification, Ontodia.org - to visualize and publish results in a form of diagrams and GitHub - to host the repository of the project, whilst Ontodia is integrated with the last. The preliminary version of the produced ontology can be accessed at: https://github.com/k0shk/pfontology. Ontodia with GitHub integration capabilities is fully operational and can be tested here: http://www.ontodia.org.

**Keywords:** semantic data visualization, collaborative ontology development, ontology development tool set, ontology production tools

## 1 Introduction

The main objective of our project was to transform large amounts of spreadsheet data in the field of budgeting available on the web to RDF linked data in order to make it accessible for cross analysis and federated queries [3]. We realized that in order to succeed one needs to have a conceptual model to apply to the transformation. One of the most critical tasks was to produce an ontology to represent the knowledge accumulated in open budget domain. The extra complications to overcome was that our team is distributed between Russia and Great Britain and not all experts are familiar with Semantic Web standards. We assume that many ontology development teams are facing the same challenges and the solution that we introduce can be beneficial and possesses a sufficient scalability.

As for ontology engineering methodology, the team has chosen to work according to METHODOLOGY: it supports re-usability and suggests a clear life cycle with evolving prototype model, thus ensuring effective process for distributed teams.

### 1.1 Workflow scenario

At the initiation stage our team adopted the following workflow scenario:

1. Collecting the appropriate data sets of open budget data.
2. Analysis of collected data and production of generalized concept maps.
3. Construction of domain model fragments in sketches and drawings.
4. Development of the ontology in authoring tool.
5. Community review and approval.

On Stages 1, 2 and 3 selection of tools is quite trivial and stays beyond the scope of the demo. For Stages 4 and 5 we prescribed the tool category with regard to the deliverables that we committed to ship (see Table 1). On the next step for each tool category we specified the requirements based on deliverables and team limitations (see Table 2).

**Table 1.** Stage goals and tool categories

| Stage | Goal | Deliverables | Tool category |
|---|---|---|---|
| 4. | To assemble the ontology in ontology authoring tool. | OWL file, slides with fragments of ontology. | Ontology authoring tool.<br>Ontology visualization tool.<br>Online repository hosting service. |
| 5. | To receive a review from the community. | Documentation for the ontology and slides. | Ontology visualization tool.<br>Online repository hosting service. |

### 1.2 Requirements justification

*Free-to-use.* Our project is non-profit and community driven one and therefore we cannot afford any pay-for products at the present stage.
*Publicity and visibility.* We are strongly interested in contributions, for which the openness is crucial.
*Support for collaborative work.* We as a team exist in different time zones and work remotely from each other and hence have a strong need for collaborative

**Table 2.** Tool categories and requirements

| Tool category | Requirement |
|---|---|
| Ontology authoring tool. | 1) Free to use; 2) Stable and reliable; 3) Supporting all basic OWL versions; 4) Rich import/export capabilities; 5) Well known to most user or easy to learn |
| Ontology visualization tool. | 1) Free to use; 2) Simple and web-based for non-coding domain experts; 3) Enables access control and publishing of content; 4) Stable and reliable; 5) Rich import/export capabilities; 6) Supports collaborative work; 7) Integrates with repository hosting service |
| Online repository hosting service. | 1) Free to use; 2) Stable and reliable; 3) Enables access control and publishing of content |

work solutions.

*Diagrams as the main artifact.* Since our team members vary greatly in their expertise and specialization the only common language they can use is diagramming of the domain knowledge.

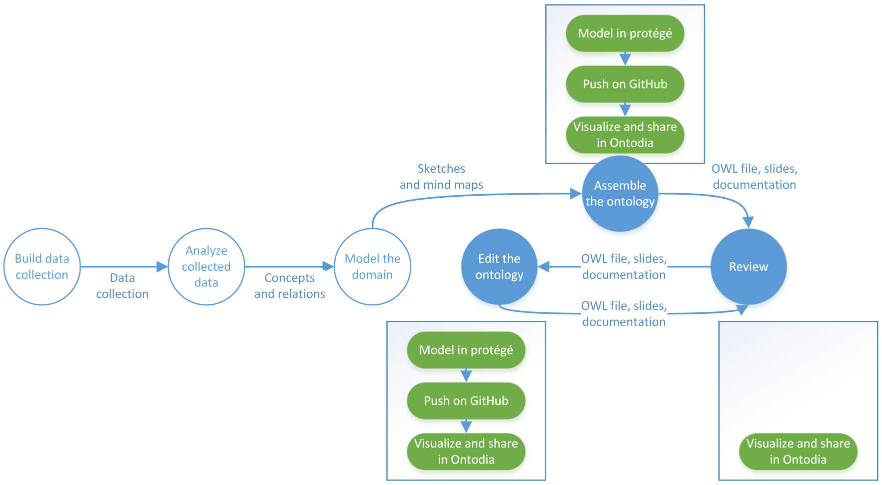


**Fig. 1.** Workflow and the choice of corresponding tools

## 2 Technical Solution

### 2.1 Tool selection

*Ontology authoring tool.* Initially the most preferable candidate was WebProtégé due to its support for collaborative work and online accessibility of its projects and files. The testing revealed that WebProtégé constituted too much effort to migrate to from offline solution since it has some peculiarities related to sign in and assignment of URI's to objects and we rolled back to limited usage of the offline Protégé. Due to Protégé being not precisely suited for generation of diagrams from ontologies and publishing on the web we still required a separate tool to visualize and publish ontologies for domain experts and community. The presence of Protégé in the stack also forced us to couple it with online repository hosting service for us to be able to exchange data and version control the content.

*Ontology visualization tool.* The tool selection is quite diverse for this task and we made use of the thorough and most detailed survey of available visualization tools [1] and found no particular tool to fulfill our vision. We made a commitment to use what we already had at our disposal: one of the team members was the contributor to ontology visualization tool development project for ITMO University. The tool was already demonstrated at ISWC2015 under the name ”Ontodia” and collected good feedback [2]. Since the tool met most of the requirements except for integration with GitHub and we entirely maintain its code, we decided to extend its functionality with integration means. The description of the integration solution is provided in the next subsection.

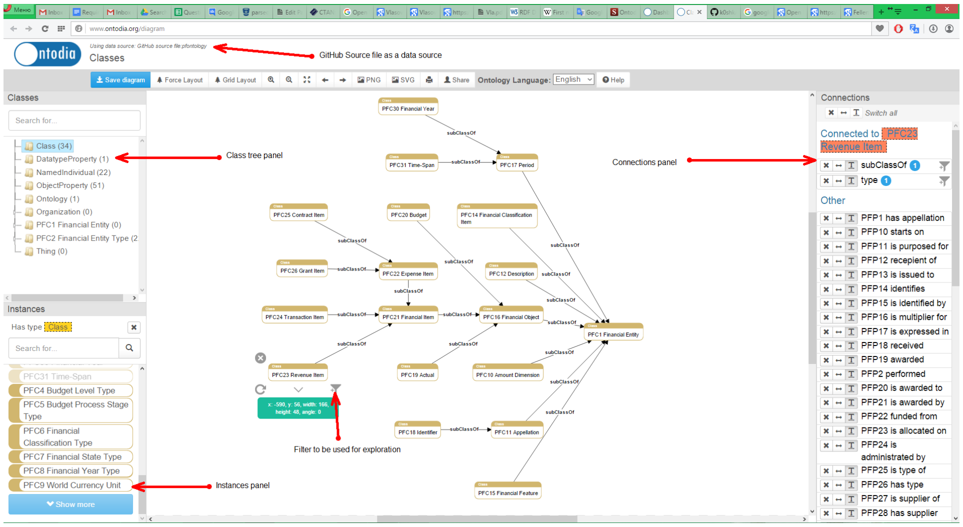
*Online repository hosting service.* Provided our requirements regarding visibility and publicity we employed GitHub to serve as our file publishing medium. The need for it arose from the decision to have Protégé as our only option for modelling environment. The presence of GitHub also resulted in additional requirement for Ontodia to be integrated with GitHub.

### 2.2 Integration of Ontodia with GitHub

The introduced integration solution is rather obvious. It was tested not only with GitHub but with WebProtégé as well. It was discovered that Github generates the permanent URL of the file page with the following structure:
https://github.com/repo owner/repo name/blob/branch name/folder/subfolder/filename. In order to get access to the file itself one needs to have the link of the following structure:
https://raw.githubusercontent.com/repo owner/repo name/repo branch/folder/subfolder/filename. The desired transformation was made with a simple regular expression operation.

Ontodia has a feature of a file upload already implemented, therefore once the link leading to the file is provided Ontodia can use it as a new data source for building diagrams upon it. From the user perspective we created a new type of data source, which we tagged ”GitHub source file” and that can be configured by providing a link to one or several files.

The other important feature for integration with any type of online data source is synchronization. We implemented two means of syncing with GitHub: (i) forced sync - it is when the user knows that some changes were made to the source file and wants to have the updated version in Ontodia, so he initiates the file update and (ii) regular sync - Ontodia once in 30 minutes syncs with the latest version of the file.



GitHub Source file as a data source

Class tree panel

Connections panel

Filter to be used for exploration

Instances panel

**Fig. 2.** Visualization of a part of the open budget ontology in Ontodia from Github OWL file

As a result the user can connect Ontodia to GitHub ontology, visualize all of it or its certain part, share it with his colleagues via their email addresses, publish it with a permanent link on the web. Ontodia can be used for preparing presentation slides by utilizing its bitmap and vector file export feature. The user invited to view the published ontology may explore the data with the use of filter button located underneath each node - See Fig.2.

Fig. 1 illustrates the proposed and well-tested workflow solution for collaborative ontology development with support for visibility and iterative approach.

The proposed toolset covers the full cycle of ontology production: a) making changes to the ontology in Protégé; b) pushing new file version on GitHub; c) obtaining and publishing automatically updated diagrams in Ontodia.

*Acknowledgements*: This work was partially financially supported by Government of Russian Federation, Grant 074-U01.

## References

1. Dudáš, M., Zamazal, O., Svátek, V.: Roadmapping and navigating in the ontology visualization landscape. In: Knowledge Engineering and Knowledge Management, pp. 137–152. Springer (2014)
2. Mouromtsev, D., Pavlov, D., Emelyanov, Y., Morozov, A., Razdyakonov, D., Galkin, M.: The simple, web-based tool for visualization and sharing of semantic data and ontologies (2015)
3. Vlasov, V., Parkhimovich, O.: Development of the open budget format. In: Open Innovations Association (FRUCT16), 2014 16th Conference of. pp. 129–136. IEEE (2014)