PROCEEDINGS OF TDWG

Biodiversity Information Science and Standards 1: e20300
doi: 10.3897/tdwgproceedings.1.20300

OPEN ACCESS

Conference Abstract

Toward a Biodiversity Data Fitness for Use Backbone (FFUB): A Node.js module prototype

Allan Koch Veiga[‡], Antonio Mauro Saraiva[§]

‡ University of São Paulo, São Paulo, Brazil
§ Universidade de São Paulo, São Paulo, Brazil

Corresponding author: Allan Koch Veiga (allan.kv@gmail.com), Antonio Mauro Saraiva (saraiva@usp.br)

Received: 14 Aug 2017 | Published: 14 Aug 2017

Citation: Veiga A, Saraiva A (2017) Toward a Biodiversity Data Fitness for Use Backbone (FFUB): A Node.js module prototype. Proceedings of TDWG 1: e20300. https://doi.org/10.3897/tdwgproceedings.1.20300

Abstract

Introduction: The Biodiversity informatics community has made important achievements regarding digitizing, integrating and publishing standardized data about global biodiversity. However, the assessment of the quality of such data and the determination of the fitness for use of those data in different contexts remain a challenge. To tackle such problem using a common approach and conceptual base, the TDWG Biodiversity Data Quality Interest Group - BDQ-IG (https://github.com/tdwg/bdq) has proposed a conceptual framework to define the necessary components to describe Data Quality (DQ) needs, DQ solutions, and DQ reports. It supports a consistent description of the meaning of DQ in specific contexts and how to assess and manage DQ in a global and collaborative environment Veiga 2016, Veiga et al. 2017. Based on the common ground provided by this conceptual framework, we implemented a prototype of a Fitness for Use Backbone (FFUB) as a Node.js module (https://nodejs.org/api/modules.html) for registering and retrieving instances of the framework concepts.

Material and methods: This prototype was built using Node.js, an asynchronous event-driven JavaScript runtime, which uses a non-blocking I/O model that makes it lightweight and efficient to build scalable network applications (https://nodejs.org). In order to facilitate the reusability of the module, we registered it in the NPM package manager (https://www.npmjs.com). To foster collaboration on the development of the module, the source code was made available in the GitHub (https://github.com) version control system. To test

the module, we have developed a simple mechanism for measuring, validating and amending the quality of datasets and records, called BDQ-Toolkit. The source code of the FFUB module can be found at https://github.com/BioComp-USP/ffub. Installing and using the module requires Node.js version 6 or higher. Instructions for installing and using the FFUB module can be found at https://www.npmjs.com/package/ffub.

Results: The implemented prototype is organized into three main types of functions: registry, retrieve and print. Registry functions enable the creation instances of concepts of the conceptual framework, as illustrated in Fig. 1, such as use cases, information elements, dimensions, criteria, enhancements, specifications, mechanisms, assertions (measure, validation, and amendment) and DQ profiles. As a prototype, these instances are not persisted, but they are stored in an in-memory JSON object. Retrieve functions are used to get instances of the framework concepts, such as DQ reports, based on the in-memory JSON object. Print functions are used to write in the console the concepts stored in the in-memory JSON object in a formatted way. Inside the FFUB module, we implemented a test which registers a set of instances of the framework concepts, including a simple DQ profile, specifications and mechanisms and a set of assertions applied to a sample dataset and its records. Based on these registries, it is possible to retrieve and print DQ reports, presenting the current status of DQ of the sample dataset and its records according to the defined DQ profile.

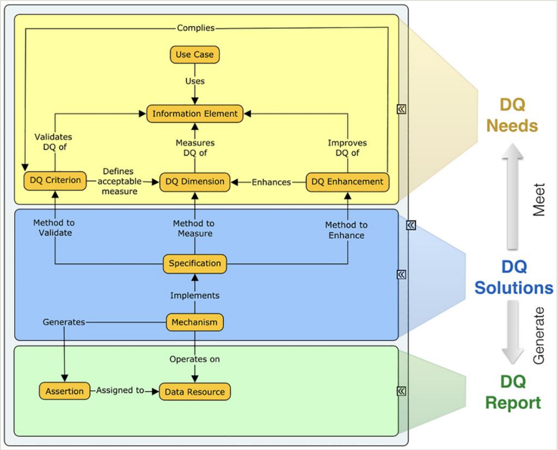

Figure 1.

The conceptual framework: Concepts and classes. DQ Needs concepts: Use Case, Information Element, DQ Dimension, DQ Criterion and DQ Enhancement. DQ Solutions concepts: Specification and Mechanism. DQ Report concepts: Data Source and Assertion (Veiga et al. 2017.)

Final remarks: This module provides a practical interface to the proposed conceptual framework. It allows the input of instances of concepts and generates, as output, information which allows the DQ assessment and management. Future work includes creating a RESTful API, based on the functions developed in this prototype, using sophisticated methods of data retrieving based on NoSQL databases.

Keywords

data quality, biodiversity data quality, fitness for use, conceptual framework

Presenting author

Allan Koch Veiga

References

- Veiga AK (2016) A conceptual framework on biodiversity data quality. Ph.D. thesis. Universidade de São Paulo, São Paulo URL: http://www.teses.usp.br/teses/disponiveis/3/3141/tde-17032017-085248/
- Veiga AK, Saraiva AM, Chapman AD, Morris PJ, Gendreau C, Schigel D, Robertson TJ (2017) A conceptual framework for quality assessment and management of biodiversity data. PloS one 12 (6): e0178731. https://doi.org/10.1371/journal.pone.0178731