In silico toxicology eNanoMapper REST services

M. Rautenberg, C. Helma, D. Gebele

in silico toxicology gmbh, Basel, Switzerland

in silico toxicology gmbh

Abstract

During the eNanoMapper project *in silico* toxicologies (IST) webservices were adapted in order to provide REST interfaces that adhere to eNanoMapper standards and specifications and supplemented with interactive SWAGGER documentation. This poster gives an overview of ISTs eNanoMapper API compatible REST services[1] and demonstrate its usage with practical examples.

lazar

lazar (lazy structure–activity relationships)[2] is a modular framework for read across predictions of chemical toxicities. Within the European Union's FP7 eNanoMapper project lazar was extended with capabilities to handle nanomaterial data and interfaces to other eNanoMapper services (databases and ontologies). lazar libraries are publicly distributed as Ruby gems.

REST

Representational State Transfer or RESTful API is an architectural style to design webservices. REST uses HTTP protocol requests GET, POST, PUT, DELETE for the communication between distributed webservices. nano-lazar[3] as a part of the eNanoMapper analysis and modelling infrastructure is built upon the OpenTox web services framework following the principles of the Representational State Transfer (REST) design model.

lazar-rest services

lazar-rest is a Ruby library, that provides a REST interface for nano-lazar. The API is compatible with eNanoMapper specifications and documented with the Swagger API framework (version 2.0), which conforms with the Open API Initiative specification.

lazar-rest Swagger interface

Swagger UI is used as an interactive documentation and visualization platform for the lazar-rest API.
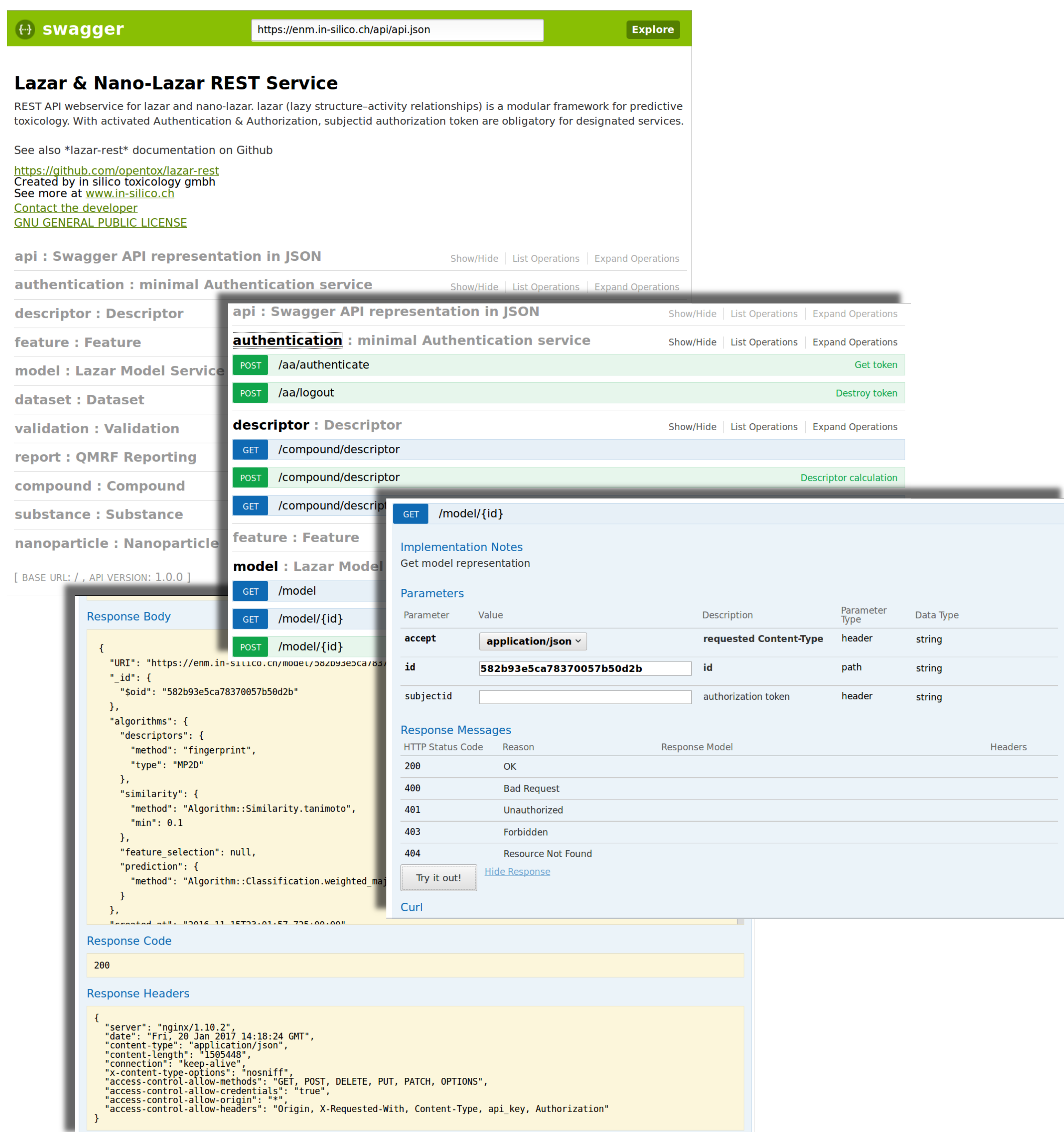

Figure 1: Swagger UI for the nano-lazar REST API

QMRF support

lazar-rest supports (Q)SAR Model Reporting Format (QMRF). The integrated QSAR-report library[4] generates QMRF reports in XML format with basic model information (e.g. validation results according to OECD guidelines). The QMRF report library is independent of other IST libraries (e.g. lazar, nano-lazar-gui, lazar-rest) and can be used by other applications that need QMRF reporting.

Integration with eNanoMapper and other services

lazar-rest provides a REST webservice for the nano-lazar framework. The lazar framework integrates internal libraries, eNanoMapper resources (Data Warehouse and Ontologies) and external resources like UniProt and BioPortal. lazar-rest services make the lazar framework accessible for other webservices and web applications and for users who prefer the command line over graphical user interfaces.
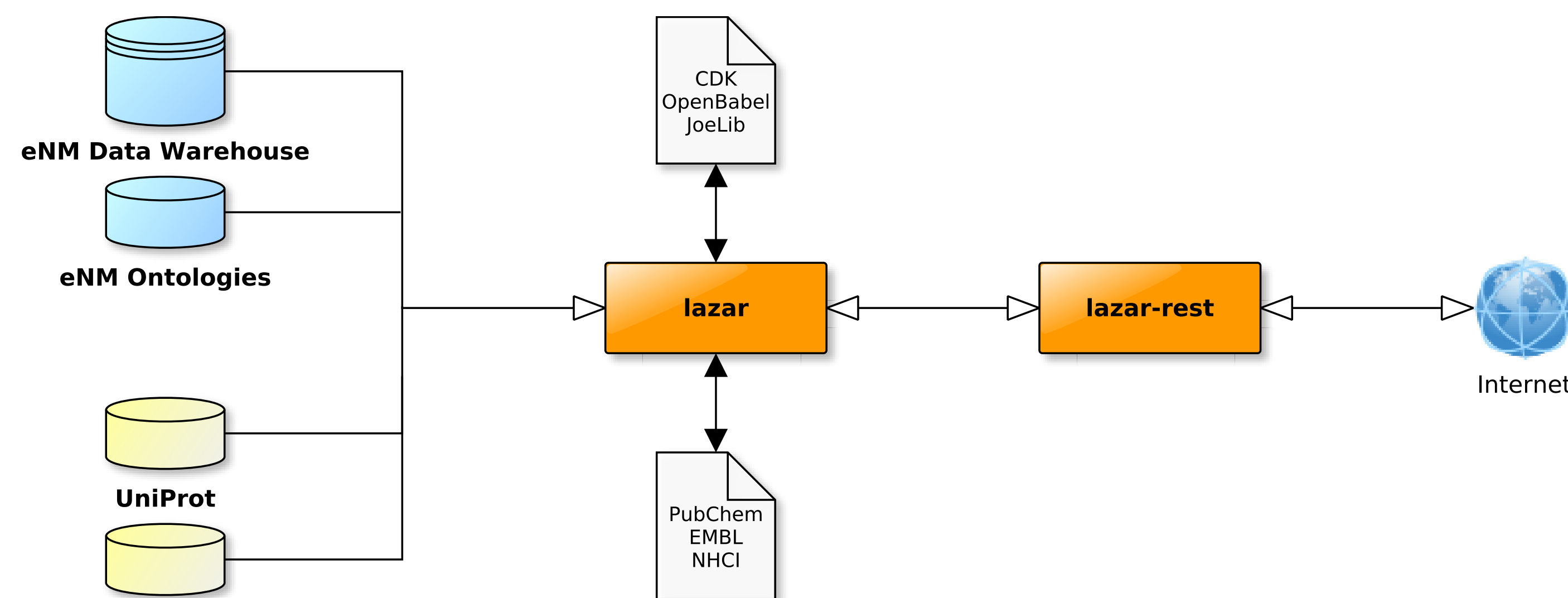

Figure 2: nano-lazar service integration

Authorization & authentication

Data confidentiality, authentication and authorization (AA) is a common requirement for many resources. Within the OpenTox[5] project, *in-silico* toxicology gmbh has implemented a single-sign-on method with an OpenLDAP backend for user management and OpenAM for authentication and authorization. This system was updated and revised for eNanoMapper and provides access control for the current services. Conforming to the OpenTox web service architecture principles[5], eNanoMapper services can be decoupled from AA. This provides the opportunity to deploy a service without AA, if required (e.g. for local installations).

Practical linux cURL command examples

Get an URI list for all models:

```
curl -X GET --header 'Accept: text/uri-list' -H 'accept:text/uri-list'
'https://enm.in-silico.ch/model'
```

Retrieve an JSON representation of a model:

```
curl -X GET -H 'Accept: application/json' -H 'accept: application/json'
'https://enm.in-silico.ch/model/<MODELID>'
```

Predict a nano-particle:

```
curl -X POST -H 'Content-Type: application/x-www-form-urlencoded' -H
'Accept:text/html' -d 'identifier=
https://enm.in-silico.ch/nanoparticle/<NANOPARTICLEID>'
'https://enm.in-silico.ch/model/<MODELID>'
```

Links

▸ nano-lazar REST API GUI. Swagger UI based documentation for API visualization and interaction.
  https://enm.in-silico.ch
▸ nano-lazar Swagger API definition file
  https://enm.in-silico.ch/api/api.json
▸ nano-lazar REST API services source code repository
  https://github.com/opentox/lazar-rest
▸ nano-lazar REST API services gem library
  https://rubygems.org/gems/lazar-rest
▸ nano-lazar REST API services Rubydocs gem library documentation
  http://www.rubydoc.info/gems/lazar-rest
▸ nano-lazar REST API issue tracker
  https://github.com/opentox/lazar-rest/issues
▸ lazar (source code) https://github.com/opentox/lazar
▸ Swagger framework for APIs http://swagger.io/
▸ Open API Initiative https://www.openapis.org/

References

[1] Micha Rautenberg, Denis Gebele, and Christoph Helma. lazar-rest, October 2016. URL https://doi.org/10.5281/zenodo.187328. source code for this version on Github: https://github.com/opentox/lazar-rest/tree/v1.0.0.

[2] Andreas Maunz, Martin Gütlein, Micha Rautenberg, David Vorgrimmler, Denis Gebele, and Christoph Helma. lazar: a modular predictive toxicology framework. *Frontiers in Pharmacology*, 4, 2013. doi: 10.3389/fphar.2013.00038. URL http://dx.doi.org/10.3389/fphar.2013.00038.

[3] Christoph Helma, Micha Rautenberg, and Denis Gebele. nano-lazar: Read across predictions for nanoparticle toxicities with calculated and measured properties address. *Frontiers in Pharmacology*, 2017. Submitted to Frontiers in Pharmacology 2017 Jan, Predictive Toxicology section.

[4] Micha Rautenberg, Christoph Helma, and Denis Gebele. qsar-report Ruby gem library, September 2016. URL https://doi.org/10.5281/zenodo.179038.

[5] Barry Hardy, Nicki Douglas, Christoph Helma, Micha Rautenberg, Nina Jeliazkova, Vedrin Jeliazkov, Ivelina Nikolova, Romualdo Benigni, Olga Tcheremenskaia, Stefan Kramer, et al. Collaborative development of predictive toxicology applications. *Journal of cheminformatics*, 2(1):1, 2010. doi: 10.1186/1758-2946-2-7.