Report from 2015 Brainhack Americas (MX)

Integration of neuroimaging results into the web

Project URL: http://nidm-api.readthedocs.org

Vanessa Sochat[1,2*] and Nolan Nichols[2]

1 Introduction

The sharing of neuroimaging research can be aided by the Neuroimaging Data Model [1] [2], a collection of specifications for documents to describe outputs of experiments, workflows, and results from neuroimaging data, software, and publications [1]. While much work is being done to integrate these standards into software and tools in the domain of human brain mapping, such a model based on the resource description framework (RDF) [3] and complex language to query it [4] makes integration of these structures into standard web development workflows highly challenging. A migration of tools for neuroimaging meta analysis [5], sharing [6], and visualization [7], into the browser comes with the implicit reality that web developers will be incentivized and able to easily integrate standards into web applications if they are provided using common, modern languages and formats. Unfortunately, Sparql has not been widely adopted [8], as the formats of choice for web development tend to be JavaScript and the JavaScript Object Notation (JSON) [9], along with flat file formats such as comma (csv) or tab separated (tsv) value text files, commonly provided by way of an Application Programming Interface (API). The goal of this Brainhack project was to develop infrastructure to serve these nidm objects, and queries to them, in modern formats to allow for the easy development of web-based tools using NIDM.

2 Approach

The `nidm-api` [10] is a web-based application that integrates these complex objects into formats that are accessible to web developers and researchers without semantic web expertise. It includes two components. First, the `nidm-api`, is a python based executable that works both as a command-line tool to run queries over the nidm data structures, as well as to serve a RESTful API to allow a local or cloud-based server to execute queries on objects accessible by URL. Second, `nidm-queries` is a repository of Sparql queries that the `nidm-api` application dynamically downloads, validates, and serves upon starting the application. This strategy means that semantic web experts can contribute to and collaborate on development of the data structures and Sparql queries without needing to worry about overall accessibility and understanding to web developers that are not knowledgeable about this niche technology. The `nidm-api`, along with serving the queries, also provides graphical web interfaces to contribute new query objects to the shared repository. By way of being a python Flask [11] application, this makes the application able to perform as both an executable to serve the API [12], along with a set of functions that can be integrated into other python based frameworks [13] or cloud platforms that provide python accessibility [14][15].

3 Results

`Using the API:` Installation produces an executable, “nidm,” that when run, downloads, validates, and provides a summary of available queries in the `nidm-queries` repository. A query can be further investigated by selecting its unique identifier:

http://localhost:8088/api/7950f524-90e8-4d54-ad6d-7b22af2e895d

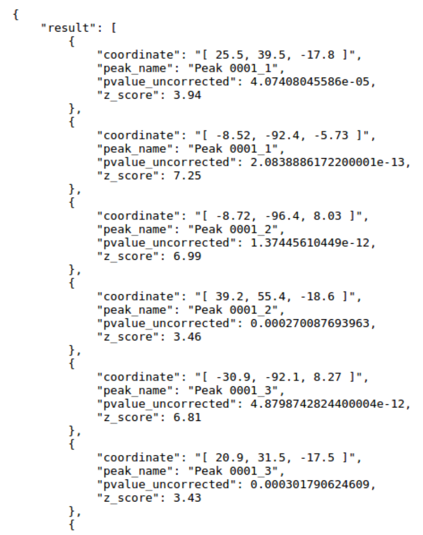
and can then be executed in a RESTful fashion by including a variable to point to a local path or URL of a nidm object:

*Correspondence: vsochat@stanford.edu
[1]Program in Biomedical Informatics, Stanford University, Stanford, 1265 Welch Road, 94306, California, USA
Full list of author information is available at the end of the article

http://localhost:8088/api/query/7950f524-90e8-4d54-ad6d-7b22af2e895d?ttl=/home/nidm.ttl

The API then runs the query over the object, and returns the result to the user in JSON 1.

The same functionality can be achieved on the command line command line in the case that it is desired to integrate it directly into a server-based python application.



```
{
    "result": [
        {
            "coordinate": "[ 25.5, 39.5, -17.8 ]",
            "peak_name": "Peak 0001_1",
            "pvalue_uncorrected": 4.07408045586e-05,
            "z_score": 3.94
        },
        {
            "coordinate": "[ -8.52, -92.4, -5.73 ]",
            "peak_name": "Peak 0001_1",
            "pvalue_uncorrected": 2.0838886172200001e-13,
            "z_score": 7.25
        },
        {
            "coordinate": "[ -8.72, -96.4, 8.03 ]",
            "peak_name": "Peak 0001_2",
            "pvalue_uncorrected": 1.37445610449e-12,
            "z_score": 6.99
        },
        {
            "coordinate": "[ 39.2, 55.4, -18.6 ]",
            "peak_name": "Peak 0001_2",
            "pvalue_uncorrected": 0.000270087693963,
            "z_score": 3.46
        },
        {
            "coordinate": "[ -30.9, -92.1, 8.27 ]",
            "peak_name": "Peak 0001_3",
            "pvalue_uncorrected": 4.8798742824400004e-12,
            "z_score": 6.81
        },
        {
            "coordinate": "[ 20.9, 31.5, -17.5 ]",
            "peak_name": "Peak 0001_3",
            "pvalue_uncorrected": 0.000301790624609,
            "z_score": 3.43
        },
        {
```

Figure 1. An example result of querying a nidm-results object (RDF) with the nidm-api returns JavaScript Object Notation (JSON).

Generating new queries: Researchers with semantic web expertise can run the application in the same fashion, and go to URL in their local browser (http://localhost:8088/query/new) to reveal an interface to generate new query objects 2. The web interface asks for a set of variables that are necessary for the `nidm-api` to serve the query. The query can be previewed, and then downloaded as a JSON object that can be submit to the `nidm-queries` repository to be added to the application.

Applications using NIDM: As an example of the utility of the NIDM standard, the NIDM results object model [16] has recently been integrated into NeuroVault NeuroVault, meaning that neuroimaging researchers can export results pertaining to statistical brain maps from common software [17] into the NeuroVault database. An nidm-viewer that runs queries over the nidm-results can then parse the coordinates and statistical parameters associated with significant locations of activations to be rendered in a table alongside a visualization of the brain map itself 3 and example. The raw data and parameters of the analysis

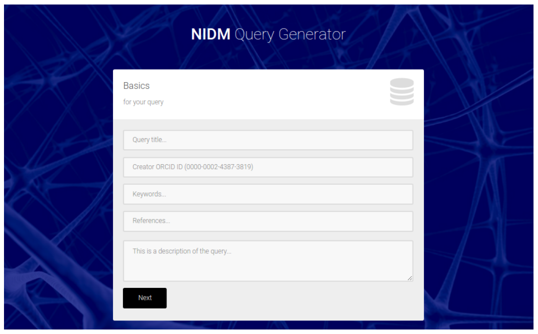
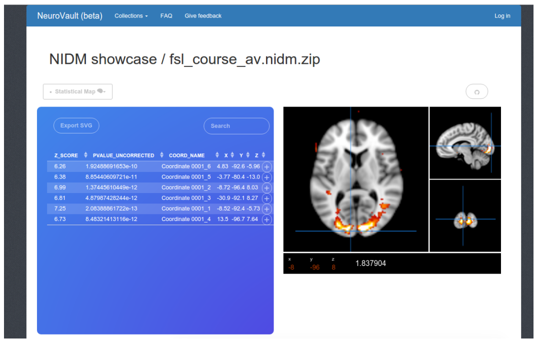


Figure 2. The tool provides an interactive web interface to contribute new queries.

are thus immediately available for sharing and publication, programatically accessible, and viewed from any web browser.



Z_SCORE	PVALUE_UNCORRECTED	COORD_NAME	X	Y	Z
6.26	1.92488691653e-10	Coordinate 0001_6	4.83	-92.6	-5.96
6.38	8.85440609721e-11	Coordinate 0001_5	-3.77	-80.4	-13.0
6.99	1.37445610449e-12	Coordinate 0001_2	-8.72	-96.4	8.03
6.81	4.87987428244e-12	Coordinate 0001_3	-30.9	-92.1	8.27
7.25	2.08388861722e-13	Coordinate 0001_1	-8.52	-92.4	-5.73
6.73	8.48321413116e-12	Coordinate 0001_4	13.5	-96.7	7.64

Figure 3. The nidm-viewer in the NeuroVault database queries nidm-result objects to generate an interactive table and statistical brain map.

4 Conclusions

By separating queries from the software to serve them, development of both can be optimized, and the nidm standard more easily deployed into tools to empower neuroimaging researchers to explore and synthesize results, workflows, and experiments. This application will be further developed to return more modern and desired outputs such as images and interactive graphs [18], and as the NIDM experiment, workflows, and results standards are further developed. The software and queries are both publicly available and open to contributions.

Availability of Supporting Data
More information about this project can be found at: `http://nidm-api.readthedocs.org`. Further data and files supporting this project are hosted in the *NIDM-nidash* repositories https://github.com/incf-nidash/nidm-api and https://github.com/incf-nidash/nidm-queries.

Competing interests
None

Author's contributions
VS and NN wrote the software and wrote the report.

Acknowledgements
The authors would like to thank the INCF-Nidash working group, along with organizers and attendees of Brainhack MX. VS is supported by the Walter Hewlett Stanford Graduate Fellowship and National Science Foundation Fellowship.

Author details
[1]Program in Biomedical Informatics, Stanford University, Stanford, 1265 Welch Road, 94306, California, USA. [2]SRI International, Menlo Park, 333 Ravenswood Ave, 94025, California, USA.

References

1. Neuroimaging Data Model Overview (NIDM-Overview). http://nidm.nidash.org/specs/nidm-overview.html. Accessed: 2015-11-24
2. NIDM Specifications. http://nidm.nidash.org/specs/. Accessed: 2015-11-3
3. RDF - Semantic Web Standards. http://www.w3.org/RDF/. Accessed: 2015-11-24
4. SPARQL Query Language for RDF. http://www.w3.org/TR/rdf-sparql-query/. Accessed: 2015-11-24
5. Yarkoni, T., Poldrack, R.A., Nichols, T.E., Van Essen, D.C., Wager, T.D.: Large-scale automated synthesis of human functional neuroimaging data. Nat. Methods **8**(8), 665–670 (2011)
6. Gorgolewski, K.J., Varoquaux, G., Rivera, G., Schwarz, Y., Ghosh, S.S., Maumet, C., Sochat, V.V., Nichols, T.E., Poldrack, R.A., Poline, J.-B., Yarkoni, T., Margulies, D.S.: NeuroVault.org: a web-based repository for collecting and sharing unthresholded statistical maps of the human brain. Front. Neuroinform. **9** (2015)
7. Research Imaging Institute — Mango. http://ric.uthscsa.edu/mango/index.html. Accessed: 2015-11-24
8. GitHut - Programming Languages and GitHub. http://githut.info/. Accessed: 2015-11-24
9. Wikipedia contributors: JSON. https://en.wikipedia.org/w/index.php?title=JSON&oldid=692109528. Accessed: 2015-11-24 (2015)
10. NIDM API — nidm 1.0 documentation. http://nidm-api.readthedocs.org/en/latest/. Accessed: 2015-11-24
11. Welcome to Flask — Flask Documentation (0.10). http://flask.pocoo.org/docs/0.10/. Accessed: 2015-11-3
12. Flask-RESTful — Flask-RESTful 0.2.1 documentation. http://flask-restful-cn.readthedocs.org/en/0.3.4/. Accessed: 2015-11-24
13. The Web framework for perfectionists with deadlines — Django. https://www.djangoproject.com/. Accessed: 2015-11-4
14. AWS Python Developer Center. https://aws.amazon.com/python/
15. Google: Python Runtime Environment. https://cloud.google.com/appengine/docs/python/. Accessed: 2015-11-24
16. NIDM-Results 1.1.0. http://nidm.nidash.org/specs/nidm-results_110.html. Accessed: 2015-11-24
17. Jenkinson, M., Beckmann, C.F., Behrens, T.E.J., Woolrich, M.W., Smith, S.M.: FSL. Neuroimage **62**(2), 782–790 (2012)
18. Neo4j, the World's Leading Graph Database. http://neo4j.com/. Accessed: 2015-11-24