Title Page

Metazen – Metadata Capture for Metagenomes

Jared Bischof[1] (jbischof@mcs.anl.gov),
Travis Harrison[1](tharriso@mcs.anl.gov),
Tobias Paczian[1] (paczian@mcs.anl.gov),
Elizabeth Glass[2] (marland@mcs.anl.gov),
Andreas Wilke[1,3] (wilke@mcs.anl.gov),
Folker Meyer[1,2,3,*] (folker@anl.gov)

AFFILIATIONS:

1. Computation Institute, University of Chicago, 5735 S Ellis Ave, Chicago, IL 60637.
2. Mathematics and Computer Science Division, Argonne National Laboratory, 9700 S. Cass Avenue, Argonne, IL 60439.
3. Institute for Genomics & Systems Biology, Argonne National Laboratory, 900 East 57th Street, Chicago, IL 60637,

* Corresponding author, Folker Meyer, folker@anl.gov

Abstract

As the impact and prevalence of large-scale metagenomic surveys grows, so does the acute need for more complete and standards compliant metadata. Metadata (data describing data) provides an essential complement to experimental data, helping to answer questions about its source, mode of collection, and reliability. Metadata collection and interpretation have become vital to the genomics and metagenomics communities, but considerable challenges remain, including exchange, curation, and distribution. Currently, tools are available for capturing basic field metadata during sampling, and for storing, updating and viewing it. Unfortunately, these tools are not specifically designed for metagenomic surveys, as they lack the appropriate metadata collection templates, a centralized storage repository, and a unique ID linking system that can be used to easily port complete and compatible metagenomic metadata into widely used assembly and sequence analysis tools.

Metazen was developed to provide a comprehensive framework to enable metadata capture for metagenomic sequencing projects. Specifically, this focuses on creating a rapid, easy-to-use portal to encourage early deposition of project and sample metadata.

Metazen is an interactive tool that aids users in recording their metadata in a complete and valid format. A defined set of mandatory fields captures vital information while the option to add fields provides flexibility.

Metazen is available at https://github.com/MG-RAST/Metazen.

Keywords

Metadata, metagenomics, collection, software

Elizabeth, I thought of a few other things. 1) We don't mention that Metazen uses the MG-RAST API to retrieve project level metadata and our metadata template. 2) Accessing these live resources (MG-RAST API and BioOntolgy) makes Metazen dynamically updated with changes from either of these. 3) Oauth authentication provides users with private project level metadata in addition to public information.

Metazen – Metadata Capture for Metagenomes

Background

As the impact and prevalence of large-scale metagenomic surveys grows, so does the acute need for more complete and standards compliant metadata. Metadata (data describing data) provides an essential complement to experimental data, helping to answer questions about its source, mode of collection, and reliability. Metadata collection and interpretation have become vital to the genomics and metagenomics communities, but considerable challenges remain, including exchange, curation, and distribution. Currently, tools are available for capturing basic field metadata during sampling, and for storing, updating and viewing it. Unfortunately, these tools are not specifically designed for metagenomic surveys, as they lack the appropriate metadata collection templates, a centralized storage repository, and a unique ID linking system that can be used to easily port complete and compatible metagenomic metadata into widely used assembly and sequence analysis tools. Metadata are frequently incomplete or are recorded using widely varying ontologies, dramatically decreasing the value of data and limiting the power of analyses. Further, correcting compatibility issues for most is a time-consuming manually performed task. Although much of the underpinning infrastructure and software already exist that could remedy this problem, the tools and technologies have not yet been brought together in a way that makes the entry, merging, and transfer of such data simple.

The Genomic Standards Consortium (GSC, http://gensc.org) has developed widely accepted metadata standards for genomic, metagenomic, and amplicon (e.g. 16S rRNA) sequence datasets [1-3]. These standards have been, and continue to be developed within the GCDML framework [2], which is both modular and extensible. The framework consists of checklists for any type of genomic data plus additional environmental packages. For example, the checklist for metagenomic data, combined with the package of metadata describing the observations of a particular environment, is a powerful means of comprehensively and consistently reporting all metadata of a particular metagenomic sample and experiment.

Capturing metadata early and in an electronic format is widely viewed as the solution to the current metadata crisis. While many software systems provide metadata capturing support (MG-RAST [4], GOLD [5], VAMPS [6], IMG/M [7], CAMERA [8], QIIME [9], IsaTools [10], RightFielder [11]), adding metadata at the time of sequence upload to an analysis resource is viewed by nearly all users as an

additional hurdle that they need to overcome as quickly as possible. The fact that less than 10% of all data sets in the MG-RAST repository have complete minimal metadata according to GSC standards highlights the nature of the crisis.

Metazen (Meta for metadata and Zen for the Japanese word for "complete") was developed to provide a comprehensive framework to enable metadata capture for metagenomic sequencing projects. Specifically, this focuses on creating a rapid, easy-to-use portal to encourage early deposition of project and sample metadata.

Implementation

Metadata is in constant flux and while many users struggle to add it to their data, as service providers, we need it to enable users to analyze their data. Think of the ability to "color" an ordination plot using user provided data (e.g. biome, sampling location, plot name, replicate name, etc.). Rather than using a single database with a fixed schema (that would be too restrictive) or a complete schema-free approach (this would be too chaotic and not offer validation), we are using a hybrid approach, combining the strength of both.

The hybrid approach enables us to capture arbitrary key value pairs and have blessed and controlled subsets of the data thus providing the best of both worlds. Figure 1 shows the general layout.

[Figure 1]

When data is uploaded (or after the fact) a separate metadata file is created. Using a simple spreadsheet (generated by Metazen in most cases), users can capture minimal metadata required for standard compliance (currently GSC MIxS).

Users can submit their metadata spreadsheet for an entire project, with many samples and extractions, for validation. These spreadsheets can also be downloaded later to modify, update, or extend the metadata.

[Figure 2]

We use two schemas, a metadata template mechanism, and a validator for controlled vocabulary. We rely on external controlled vocabularies e.g. BioOntology.org via web-services or downloaded flat files.

Schema 1: Basic data

Generalized metadata storage: This is our "capture all" storage mechanism, enabling us to capture arbitrary key value pairs. This is important in a field that is currently discovering the need for metadata acquisition and storage, and the tools to perform these tasks.

We use a very simple schema: (id | collection | key | value) to store arbitrary metadata on any data type. By (optionally) organizing metadata into collections we can for each individual collection:

- require certain fields (Keys)
- validate the use of controlled vocabulary terms (when required for certain values) for given fields.

[Figure 3]

Schema 2: A meta-metadata database
This database describes the required fields and the controlled vocabularies for each "namespace|domain". Example domains are GSC MIxS, GSC MIMS. Metazen uses the MG-RAST API to retrieve project level metadata and our metadata template. Accessing these live resources (MG-RAST API and BioOntolgy) makes Metazen dynamically updated with changes from either of these. Metazen also uses an Oauth authentication, which provides users with private project level metadata in addition to public information.

[Figure 4]

[Figure 5]

Results and Discussion

Metazen is an interactive tool that aids users in recording their metadata in a complete and valid format. A defined set of mandatory fields captures vital information while the option to add fields provides flexibility. Project and user level information is stored so users can reload that information without having to enter it repeatedly. Entry into Metazen starts with the login page (Figure 6).

[Figure 6]

Metazen presents a simple user-friendly web interface that has a limited number of mandatory terms (following the GSC standards). Project level information can be pre-filled (if desired by the user) from existing projects in the system. Figure 7 shows the fields that have been pre-filled by the stored user information, required fields versus optional fields, and a highlighted field that indicates how the user is informed of missing required fields or invalid data entry.

[Figure 7]

Figure 8 shows where users can pick an environmental package (top region of screenshot). Also, complex, controlled vocabularies are more easily navigated and chosen through Bioportal widgets. Help with the selection of controlled vocabulary terms that help explain data to third parties is facilitated via widgets imported from Bioontology.org. Users can explore the existing controlled vocabulary terms and prefill the spreadsheet.

[Figure 8]

Editing and validating metadata
Once users are finished entering their metadata into Metazen, a spreadsheet can be generated for download and further editing. Often times users will want to edit their spreadsheet manually if they have many samples and/or sequencing runs for which they would like to enter specific metadata on a per sample or per run basis. Having both the web and spreadsheet to interact with, provides users with guidance to follow standards and the flexibility to extend and modify their metadata. Once they are finished editing their metadata, we have a validation tool to help users identify metadata errors, and to ensure that the submitted spreadsheet contains the vital (required) metadata fields and completely valid data.

We have created architecture for a new, user-friendly metadata capture software application that will be accessible via the Internet and with commonly used smart phone operating systems (Android and Apple IOs) called Metazen Collect. It has the capability to easily export metadata files in Excel.

Conclusions

Metadata is in constant flux and while many users struggle to add it to their data, as service (MG-RAST) providers, we need it to enable users to analyze their data. Metazen provides researchers with a tool to collect and contribute metadata using community data standards and controlled vocabularies. Highlights and unique attributes of this software include:

- Project level information can be saved for re-use when creating the metadata for a new project.
- Required field's help to capture at a minimum the most vital information.
- Field validation at this interactive site helps to ensure the integrity of the metadata and guides the user through what is often times a more difficult endeavor.
- A strict format requirement for various data types provides the ability to search metadata fields at a later time.
- Controlled vocabularies are more easily navigated and chosen through dropdown menus and BioPortal bioontology [12] widgets.
- Searching the Google maps API can help users to obtain the geographic coordinates of where their samples were obtained.
- To assist third parties, the metadata created is available for download in MS Excel format from the MG-RAST download pages and via the web services API (Metazen uses the MG-RAST API to retrieve project level metadata and our metadata template).

The Metazen architecture allows for easy extensions into new metadata packages and new data types (like GWAS, metaproteomes or microarray data). The GSC

metadata mechanism is plug-in enabled. Domain scientists can create new required subsets of terms (and the corresponding controlled vocabulary) to capture more data on their field of study.

Availability and requirements

- **Project name:** Metazen
- **Project home page:** https://github.com/MG-RAST/Metazen
- **Operating system(s):** Platform independent
- **Programming language:** Perl, CGI, and Javascript
- **Other requirements:** *e.g. Java 1.3.1 or higher, Tomcat 4.0 or higher*
- **License:** GNU GPL
- **Any restrictions to use by non-academics:** None

List of abbreviations

API: Application Programming Interface
GCDML: Genomic Contextual Data Markup Language
GNU: GNU's Not Unix
GSC: Genomic Standards Consortium
XML: Extensible Markup Language

Competing Interests

None.

Author Contributions

J.B was the main developer for Metazen; T.H and A.W. developed the metadata collection and retrieval infrastructure in MG-RAST; T.P. implemented Metazen Collect application for mobile devices; E.M.G wrote the manuscript and evaluated UI; F.M. conceived of the project and he and A.W. provided overall and technical management, respectively.

Acknowledgements

All authors gratefully acknowledge the support of METAZEN by the Gordon and Betty Moore Foundation, Grant 3354. This work was supported in part by the Office of Advanced Scientific Computing Research, Office of Science, U.S. Dept. of Energy, under Contract DE-AC02-06CH11357.

References

1. Field D, Garrity G, Gray T, Morrison N, Selengut J, Sterk P, Tatusova T, Thomson N, Allen MJ, Angiuoli SV, Ashburner M, Axelrod N, Baldauf S, Ballard S, Boore J, Cochrane G, Cole J, Dawyndt P, De Vos P, DePamphilis C, Edwards R, Faruque N, Feldman R, Gilbert J, Gilna P, Glöckner FO, Goldstein P, Guralnick R, Haft D, Hancock D, et al.: **The minimum information about a genome sequence (MIGS) specification.** *Nat Biotechnol* 2008, **26**(5):541-7.
2. Kottmann R, Gray T, Murphy S, Kagan L, Kravitz S, Lombardot T, Field D, Glöckner FO, Genomic Standards Consortium: **A standard MIGS/MIMS compliant XML Schema: toward the development of the Genomic Contextual Data Markup Language (GCDML).** *OMICS* 2008, **12**(2):115-21.
3. Yilmaz P, Kottmann R, Field D, Knight R, Cole JR, Amaral-Zettler L, Gilbert JA, Karsch-Mizrachi I, Johnston A, Cochrane G, Vaughan R, Hunter C, Park J, Morrison N, Rocca-Serra P, Sterk P, Arumugam M, Bailey M, Baumgartner L, Birren BW, Blaser MJ, Bonazzi V, Booth T, Bork P, Bushman FD, Buttigieg PL, Chain PS, Charlson E, Costello EK, Huot-Creasy H, et al.: **Minimum information about a marker gene sequence (MIMARKS) and minimum information about any (x) sequence (MIxS) specifications.** *Nat Biotechnol* 2011, **29**(5):415-20.
4. Meyer F, Paarmann D, D'Souza M, Olson R, Glass EM, Kubal M, Paczian T, Rodriguez A, Stevens R, Wilke A, Wilkening J, Edwards RA: **The metagenomics RAST server - a public resource for the automatic phylogenetic and functional analysis of metagenomes.** *BMC Bioinformatics* 2008, **19**;9:386.
5. Pagani I, Liolios K, Jansson J, Chen IM, Smirnova T, Nosrat B, Markowitz VM, Kyrpides NC: **The Genomes OnLine Database (GOLD) v.4: status of genomic and metagenomic projects and their associated metadata.** *Nucleic Acids Res* 2012, **40**(Database issue):D571-9.
6. **VAMPS - The Visualization and Analysis of Microbial Population Structures** [http://vamps.mbl.edu/]
7. Markowitz VM, Chen IM, Chu K, Szeto E, Palaniappan K, Grechkin Y, Ratner A, Jacob B, Pati A, Huntemann M, Liolios K, Pagani I, Anderson I, Mavromatis K, Ivanova NN, Kyrpides NC: **IMG/M: the integrated metagenome data management and comparative analysis system.** *Nucleic Acids Res* 2012, **40**(Database issue):D123-9.
8. Sun S, Chen J, Li W, Altintas I, Lin A, Peltier S, Stocks K, Allen EE, Ellisman M, Grethe J, Wooley J: **Community cyberinfrastructure for Advanced Microbial Ecology Research and Analysis: the CAMERA resource.** *Nucleic Acids Res* 2011, **39**(Database issue):D546-51.
9. Caporaso JG, Kuczynski J, Stombaugh J, Bittinger K, Bushman FD, Costello EK, Fierer N, Peña AG, Goodrich JK, Gordon JI, Huttley GA, Kelley ST, Knights D, Koenig JE, Ley RE, Lozupone CA, McDonald D, Muegge BD, Pirrung M, Reeder J, Sevinsky JR, Turnbaugh PJ, Walters WA, Widmann J, Yatsunenko T, Zaneveld J, Knight R: **QIIME allows analysis of high-throughput community sequencing data.** *Nat Methods* 2010, **7**(5):335-6.
10. Rocca-Serra P, Brandizi M, Maguire E, Sklyar N, Taylor C, Begley K, Field D, Harris S, Hide W, Hofmann O, Neumann S, Sterk P, Tong W, Sansone SA: **ISA software suite: supporting standards-compliant experimental annotation and enabling curation at the community level.** *Bioinformatics* 2010, **15**;26(18):2354-6.
11. RightFielder [http://www.melissadata.com/dm/mailing-software/rightfielder.htm]

12. BioPortal bioontology [http://bioportal.bioontology.org/]

Figure Legends

Figure 1: A three tiered hierarchy allows modeling of real life experiments, combining multiple "samples" into a "project" (or study) and allowing for different "extractions" from the sample.
Figure 2: All data files are accompanied by metadata. The tuples of payload file(s) and a metadata file are uploaded together and the metadata file is potentially validated. Separate metadata files describe the data, different "schemata" described project, sample and data, however the lower tiers can inherit data from the higher tiers i.e. project can contain multiple samples.
Figure 3: The backend storage mechanism provides two different tables. MDEntry to store all data, and MDCollection to group entries into sets.
Figure 4: The meta-metadata database provides two essential tables describing structure and syntax of certain metadata. For a given metadata collection type the MDTemplate table provides information about metadata categories, value format for the category and source name (e.g. MIAME). MDControlledVocabulary contains valid values for defined keys with type CV in MDTemplate
Figure 5: Using MDTemplate and MDCV we enable checking for required keys and controlled vocabulary terms.
Figure 6: Metazen login page.
Figure 7: Metadata form. Stored information can be used to prefill fields. Not all fields are required, but they are validated. Please note in this example the PI Organization is a required field that was left blank.
Figure 8: Viewing controlled vocabularies in Metazen.