LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

UCRL-TR-236531

Viral Metagenomics: MetaView Software

C. Zhou, J. Smith

November 14, 2007

Disclaimer

This document was prepared as an account of work sponsored by an agency of the United States government. Neither the United States government nor Lawrence Livermore National Security, LLC, nor any of their employees makes any warranty, expressed or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States government or Lawrence Livermore National Security, LLC. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States government or Lawrence Livermore National Security, LLC, and shall not be used for advertising or product endorsement purposes.

This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

LLNL FY07 Tech-base Project, “Viral Metagenomics: MetaView Software”
Carol Zhou, Jason Smith
Report submission date: 22 October 2007
Funding allocation: $50k
Project duration: Jan-Sept. 2007
UCRL-TR-236531

Purpose. To design and develop a tool for analysis of raw sequence read data from viral metagenomics experiments. The tool should compare read sequences of known viral nucleic acid sequence data and enable a user to attempt to determine, with some degree of confidence, what virus groups may be present in the sample.

Abstract. This project was conducted in two phases. In phase 1 we surveyed the literature and examined existing metagenomics tools to educate ourselves and to more precisely define the problem of analyzing raw read data from viral metagenomic experiments. In phase 2 we devised an approach and built a prototype code and database. This code takes viral metagenomic read data in fasta format as input and accesses all complete viral genomes from Kpath for sequence comparison. The system executes at the UNIX command line, producing output that is stored in an Oracle relational database. We provide here a description of the approach we came up with for handling un-assembled, short read data sets from viral metagenomics experiments. We include a discussion of the current MetaView code capabilities and additional functionality that we believe should be added, should additional funding be acquired to continue the work.

Background Information. Metagenomics analysis is a relatively new field within bio-informatics. For that reason, there are rather few tools available for analysis of metagenomic data, let alone data from viral metagenomics experiments in particular. The difficulties in analyzing bacterial metagenomics experiments are evident. Determining what organisms are present in a given sample (“library”) is like piecing together several (or more, perhaps by orders of magnitude) jig-saw puzzles in which the pieces have been thrown together. Compounding this problem is the dominance of species that may be completely unknown, and the reality we may only have sequence information for a tiny fraction of species from the highly diverse microbial world.

But these difficulties seem almost trivial in comparison to those faced by researchers trying to identify viruses. Viruses are intracellular parasites, meaning that they complete their life cycle within cells of microbes or higher organisms. It is estimated that each bacterial species may be infected by tens to hundreds of viral species—thus, the diversity within the viral world is enormous. Furthermore, virus taxonomy is by no means a mature science. Attempts to base viral taxonomy on morphology are confounded by the high frequency of lateral gene transfer among viruses. Although the ICTV virus taxonomy is today’s standard, alternate approaches to virus taxonomy are being developed. As an alternate approach, the Rohwer lab, for example, is developing a new method based on clustering and identification of “signature” genes. However, it is often difficult even to identify a nucleic acid sequence as viral, because ~60% of viral sequences are completely novel, and many others resemble bacterial sequences. There are numerous obstacles to overcome in building a viral metagenomic analysis tool that will provide reliable and meaningful results. Our conclusion from the literature survey was that the tool we endeavored to build would represent a very partial solution to the problem of describing what known viral groups (at unspecified levels within the taxonomic hierarchy) may be detectable, but that the tool might find its greatest utility in the hands of a researcher wishing to survey the data from a particular angle.

The References section lists the research articles that were consulted for this background information. Additional notes regarding these references will be provided upon request.

Tools Review. The following tools were examined for their suitability in this project: PHACCS, Phage Proteomic Tree, Viral Genome Database, VirGen, NCBI virus database, ICTV virus taxonomy, JGI’s IMG, Rohwer lab tools (blast, cironspect, FastBlast, FastGroup, FastView, GetSequence, Mapping, Phage Database, SCUM). Notes regarding these reviews will be provided upon request.

System Overview. We decided to build a tool that would map individual sequence reads to specific virus groups. Each group would be defined by an NCBI reference sequence and a set of closely related complete viral genome neighbors, which we decided to call a “focus group”. A focus group corresponds roughly to the taxonomic levels of species, genus, or family, depending on the depth of representation of the refseqs in Genbank. We reasoned that individual reads too short to assemble, or comprising sequence from organisms with high rates of genomic variability, could not reasonably be mapped to individual genes of known organisms represented in Genbank. However, we reasoned that if among a large set of sequence reads, there were a preponderance of Blast hits to a given focus group, then one might postulate that a member of that focus group is present in the library. Fig. 1 illustrates the method we devised for approaching the problem of identifying the presence of a taxonomic group within a metagenomic sample, and breaks the development effort into 4 logical steps.

In step 1 we build the focus groups. Each complete genome is mapped to its closest refseq based on taxonomic proximity. The set of focus groups then comprise a set of blast databases against which viral metagenomic reads can be Blasted. Thus, a blast hit against a genome neighbor can be translated to its corresponding position on the reference sequence. The focus groups and their mappings are stored in the metaseq database. In step 2 we blast a set of viral metagenomic reads against the library of focus groups (i.e., against each genome sequence in each focus group). The results of these blasts are stored in the metaseq database. In step 3, we prepare various scripts for performing “canned” queries across the metaseq data set. Our relational schema was designed to facilitate data mining in the sense that we provide the user the capability of gathering aggregate statistical data arising from the raw blast hit data. For example, the user might wish to determine which focus groups incurred the most hits to answer the question, “which virus groups may be most represented”; or s/he may wish to examine a particular focus group to determine the hit density (in depth or in breadth). Although the focus group is a kind of “pseudo taxon”, one could use the metaseq data set to transcend the taxonomic tree and sum up the hits within a given family, for example. Many such user-defined scenarios are possible, enabled by intelligent queries within our relational schema. In step 4 we provide a user-friendly web interface whereby the user could query using canned scripts, visualize results, and hyperlink to detailed information, successively drilling down all the way to individual blast hits, if desired.

In this project we implemented steps 1 and 2. Due to time/effort limitations, we recognized the functionality that we would have liked to implement, and we prioritized according. Below is specific information about what the tool does and does not do.

System Design.

1) Database. We designed and implemented an Oracle relational database called “metaseq” (Fig. 2) for holding MetaView results sets.

2) Software. The software base and components for the MetaView project were based upon a true Object Oriented Programming approach using software engineering techniques. The project is divided up into three sub packages, one of which serves as a subsystem to the other two packages. The subsystem called SimilaritySearch was designed to be a basal level and generic sequence comparison suite. Currently, the software suite supports BLAST as the only sequence search capability, but future search methods can be added to the SimilaritySearch package with minimal impact to existing code, if any. The other two software packages support the two principle functionality requirements of the MetaView project which are Mapping genome sequences to a reference sequence (MetaView::Map), and comparing metagenomic data against the pre-mapped genome sequence groups for coverage calculations (MetaView::Read).

For the MetaView::Map package, a key abstraction is the FocusGroup object which holds information about individual sequences that are related through proximity on the taxonomy tree at NCBI. In this package, there is a handler that serves as a proxy to this key abstraction, the SimilaritySearch package and the MetaView database client object for storing persistent data. The MetaView::Read package is designed in a similar manner but has one key difference. The members of the key abstraction in the MetaView:Read package, a MetaExperiment, can have its SimilaritySearch results associated with a FocusGroup object from the MetaView::Map package. This allows for the metagenomic comparison results to be queried faster for the Graphical User Interface to display the resultant data.

The software for a user interface presentation layer has not been developed outside of the initial test scripts used to verify functionality of the object oriented system.

Source Codes. The software has been installed globally under /usr/local/bin and can be used by PERL code with inclusion of a “use MetaView” statement.

Capability.

<u>What the prototype software does:</u>
- construct focus groups by associating each complete viral genome with its closest refseq
- process raw reads, mapping them onto genomes in the closest focus group
- store low-level blast information for all match analyses (focus group construction and sequence read mapping)
- implement an Oracle relational data store (although data are currently stored in Unix flat files)

<u>What the prototype software does not do:</u>
- extract genomes from NCBI (currently uses Kpath database)
- bundle multiple segments for segmented viruses
- read/write to the relational database
- synthesize low-level data to produce summary information (e.g., sequence read hit density (depth or breadth) per focus group)
- aggregate data below arbitrary node of focus group hierarchy
- present functionality by means of a user-friendly web interface there is no versioning

Concluding Remarks. In this project we studied the problem of viral meta-genomic sequence read analysis and devised a reasonable approach to the annotation of un-assembled reads. In the time/effort allotted, we made good progress toward implementing a software and database that would be useful as a research tool. We have produced a prototype system that, with some additional development, could be deployed in viral metagenomic analysis. However, a the full utility that could be achieved would require a significant effort; we suggest that this tool could be presented as the basis for an expanded Tech-base or sponsor-funded project.

References.

Angly F, Rodriguez-Brito B, Bangor D, McNairnie P, Breitbart M, Salamon P, Felts B, Nulton J, Mahaffy J, Rohwer F: PHACCS, an online tool for estimating the structure and diversity of uncultured viral communities using metagenomic information. BMC Bioinformatics 2005, 6:41 doi:10.1186/1471-2105-6-41.

Breitbart M, Felts B, Kelley S, Mahaffy JM, Nulton J, Salamon P, Rohwer F: Diversity and population structure of a near-shore marine-sediment viral community. Proceedings of the Royal Society of London. Biological Sciences 2004, 271:565-574.

Breitbart M, Hewson I, Felts B, Mahaffy JM, Nulton J, Salamon P, Rohwer F: Metagenomic analyses of an uncultured viral community from human feces. Journal of Bacteriology 2003, 185:6220-6223, doi:10.1128/JB.185.20.6220-6223.2003.

Breitbart M, Miyake JH, Rohwer F: Global distribution of nearly identical phage-encoded DNA sequences. FEMS Microbiology Letters 2004, 236:249-256.

Breitbart M and Rohwer F: Method for discovering novel DNA viruses in blood using viral particle selection and shotgun sequencing. BioTechniques 2005, 39:729-736.

Breitbart M and Rohwer F: Here a virus, there a virus, everywhere the same virus? Trends in Microbiology 2005, 13; doi:10.1016/j.tim.2005.04.003.

Cann A J, Fandrich ES, Heaphy S: Analysis of the virus population present in equine faeces indicates the presence of hundreds of uncharacterized virus genomes. Virus Genes 2005, 30:151-156

Edwards R: Random Community Genomics. April 2006. White paper.

Edwards RA, Rohwer F: Viral metagenomics. Nature 2005, 3:504-510.

Ferris MM, Yoshida TM, Marrone BL, Keller RA: Fingerprinting of single viral genomes. Analytical Biochemistry 2005, 337:278-288.

Hiscock D, Upton C: Viral Genome DataBase: storing and analyzing genes and proteins from complete viral genomes. Bioinformatics 2000, 16:484-485.

Kulkarni-Kale U, Bhosle S, Manjari GS, Kolaskar AS: VirGen: a comprehensive viral genome resource. Nucleic Acids Research 2004, 32:D289-292.

Lawrence JG, Hatfull GF, Hendrix RW: Imbroglios of viral taxonomy: genetic exchange and failing of phonetic approaches. Journal of Bacteriology 2002, 184:4891-4905.

Martin et al.: Metagenomic analysis of two enhanced biological phosphorus removal (EBPR) sludge communities. Nature Biotechnology 2006, 24:1263-1269.

Martin HG, Ivanova N, Kunin V, Warnecke F, Barry KW, McHardy AD, Yeates C, He S, Salamov AA, Szeto E, Dalin E, Putnam NH, Shapiro JH, Pangilinan JL, Rigoutsos I, Dyrpides NC, Blackall LL, McMahon KD, Hugenholtz P: Metagenomic analysis of two enhanced biological phosphorous removal (EBPR) sludge communities. Nature Biotechnology 2006, 24:1263-1269.

Nelson D: Phage taxonomy: we agree to disagree. Journal of Bacteriology 2004, 186:7029-7031.

Rohwer E, Edwards R: The phage proteomic tree: a genome-based taxonomy for phage. Journal of Bacteriology 2002, 184:4529-4535.

Tyson et al.: Community structure and metabolism through reconstruction of microbial genomes from the environment. Nature 2004, 428:37-43.

Tringe et al: Comparative metagenomics of microbial communities. Science 2005, 308:554-557.

Virus Taxonomy: Classification and Nomenclature of Viruses. Eighth Report of the International Committee on Taxonomy of Viruses (book). Ed.C.M. Fauquet, M.A. Mayo, J. Maniloff, D.J., U. Desselberger and L.A. Ball (2005). Academic Press, San Diego

Woyke et al.: Symbiosis insights through metagenomic analysis of a microbial consortium. Nature 2006, doi:10.1038/nature05192.

Xu J: Microbial ecology in the age of genomics and metagenomics: concepts, tools, and recent advances. Molecular Ecology 2006, 15:1713-1731.

Zhang T, Breitbart M, Lee WH, Run J-Q, Wei CL, Soh SWL, Hibberd ML, Liu ET, Rohwer F, Ruan Y: RNA viral community in human feces: prevalence of plant pathogenic viruses. PLoS Biology 2006, 4:0108-0118.

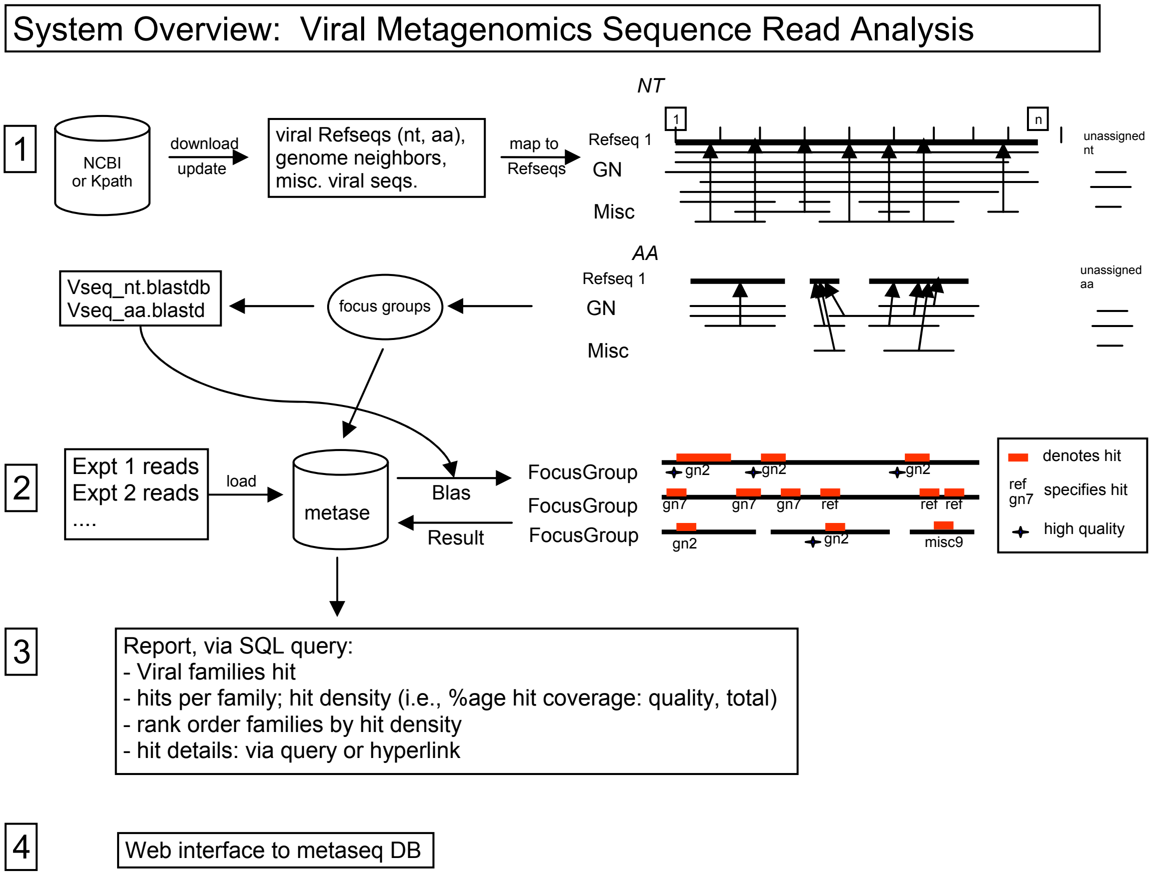
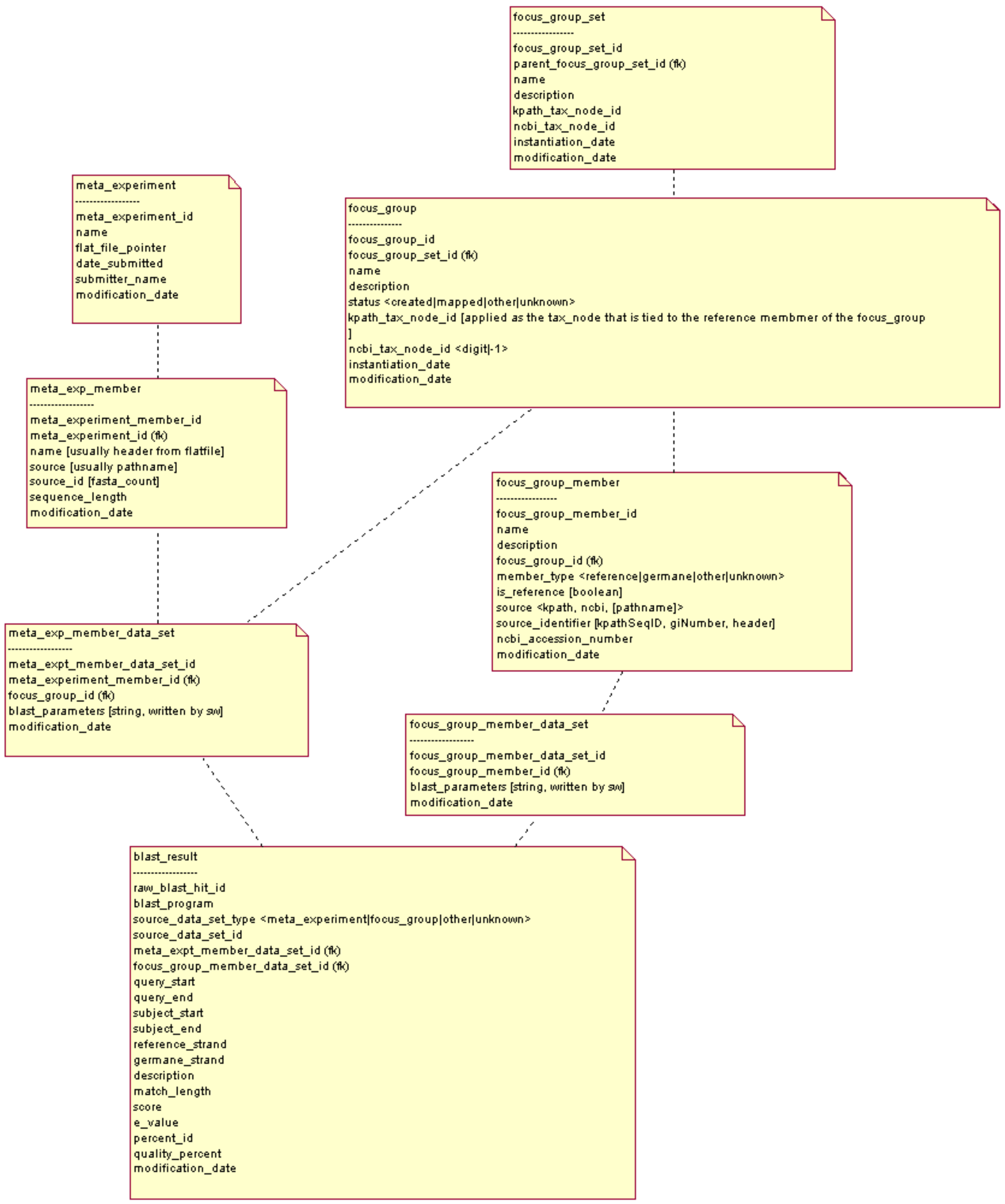


Fig. 1. Conceptual design for 4-phase development of MetaView software.



```
focus_group_set
-----------------
focus_group_set_id
parent_focus_group_set_id (fk)
name
description
kpath_tax_node_id
ncbi_tax_node_id
instantiation_date
modification_date
```

```
meta_experiment
------------------
meta_experiment_id
name
flat_file_pointer
date_submitted
submitter_name
modification_date
```

```
focus_group
---------------
focus_group_id
focus_group_set_id (fk)
name
description
status <created|mapped|other|unknown>
kpath_tax_node_id [applied as the tax_node that is tied to the reference membmer of the focus_group
]
ncbi_tax_node_id <digit|-1>
instantiation_date
modification_date
```

```
meta_exp_member
------------------
meta_experiment_member_id
meta_experiment_id (fk)
name [usually header from flatfile]
source [usually pathname]
source_id [fasta_count]
sequence_length
modification_date
```

```
focus_group_member
-----------------
focus_group_member_id
name
description
focus_group_id (fk)
member_type <reference|germane|other|unknown>
is_reference [boolean]
source <kpath, ncbi, [pathname]>
source_identifier [kpathSeqID, giNumber, header]
ncbi_accession_number
modification_date
```

```
meta_exp_member_data_set
------------------
meta_expt_member_data_set_id
meta_experiment_member_id (fk)
focus_group_id (fk)
blast_parameters [string, written by sw]
modification_date
```

```
focus_group_member_data_set
------------------
focus_group_member_data_set_id
focus_group_member_id (fk)
blast_parameters [string, written by sw]
modification_date
```

```
blast_result
------------------
raw_blast_hit_id
blast_program
source_data_set_type <meta_experiment|focus_group|other|unknown>
source_data_set_id
meta_expt_member_data_set_id (fk)
focus_group_member_data_set_id (fk)
query_start
query_end
subject_start
subject_end
reference_strand
germane_strand
description
match_length
score
e_value
percent_id
quality_percent
modification_date
```

Fig. 2. Metaseq relational database schema.