KBase Performance Metric for FY17: Develop improved open access platforms for computational analysis of large genomic datasets.

Q1 Metric: Report on the latest capabilities for annotating and assembling genome-based metabolic models of microbial metabolism.

Introduction

A major challenge in this data-rich age of biology is integrating heterogeneous, distributed, and error-prone primary and derived data into predictive models of biological function ranging from a single gene to entire organisms and their ecologies. The DOE Systems Biology Knowledgebase (KBase, www.kbase.us) is a software platform designed to make it easier for scientists to create, execute, collaborate on, and share sophisticated, reproducible analyses of their biological data in the context of public data and data other users have privately shared with them [1]. KBase is open access and free for anyone to use.

KBase supports a growing and extensible set of applications (apps) for contig assembly, genome annotation, metabolic model reconstruction, flux balance analysis, expression analysis, and comparative genomics. In addition to these tools, the KBase platform provides data integration and search, along with easy access to shared user analyses of public plant and microbial reference data.

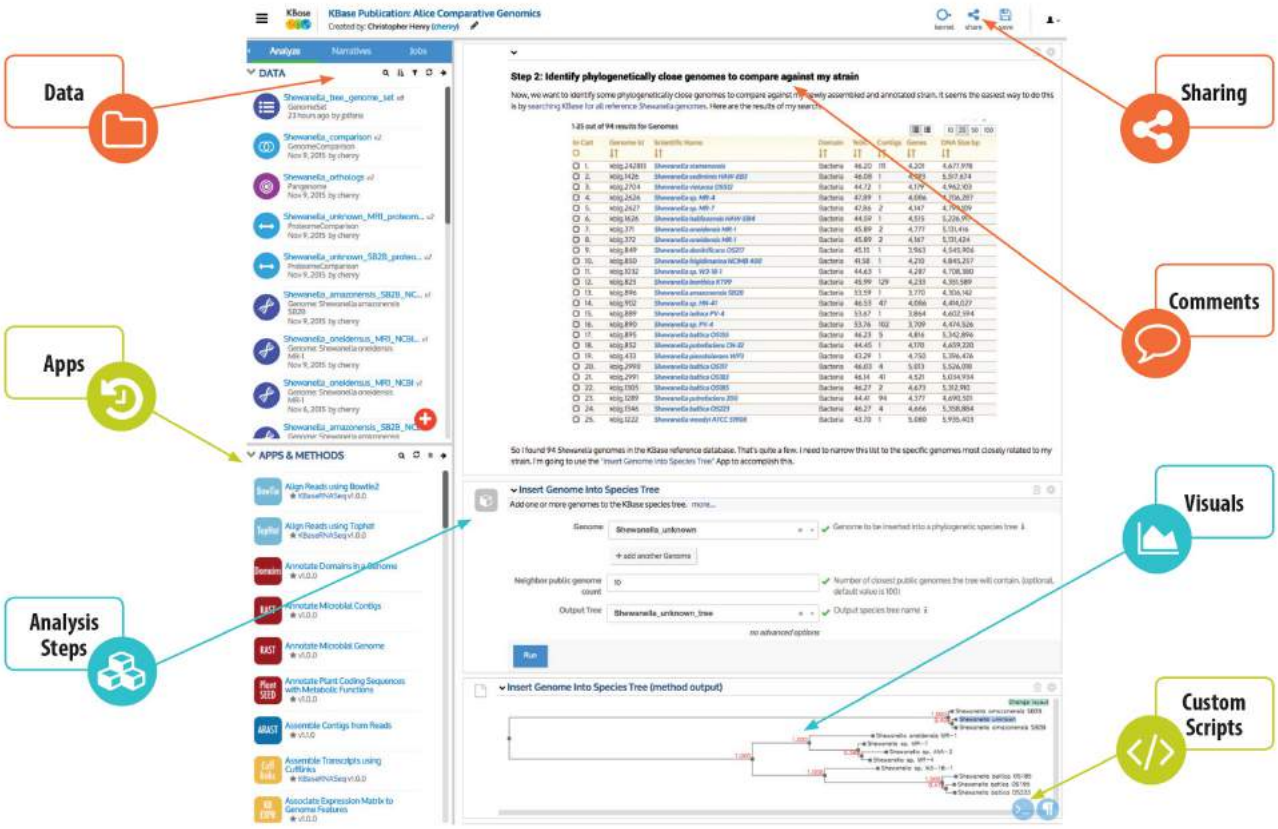


Figure 1. *KBase Narrative. A Narrative is an interactive, dynamic, and persistent document created by users that promotes open, reproducible, and collaborative science.*

KBase's graphical user interface supports both point-and-click and scripting access to system functionality in a "notebook" environment. Built on the Jupyter Notebook [2], the interface allows researchers to design, carry out, record, and share computational experiments in the form of Narratives—dynamic, interactive documents that include all the data, analysis steps, parameters, visualizations, scripts, commentary, results, and conclusions of an experiment (Figure 1).

KBase users have applied the system to address a range of scientific problems, including comparative genomics of plants, prediction of microbiome interactions, and deep metabolic modeling of environmental and engineered microbes. The Narratives they have chosen to share publicly (see http://kbase.us/narrative-library) can be viewed, copied, and re-run, possibly with different parameters or new datasets. By enabling reproducible and reusable scientific analysis and facilitating collaboration, KBase is accelerating the pace of systems biology research.

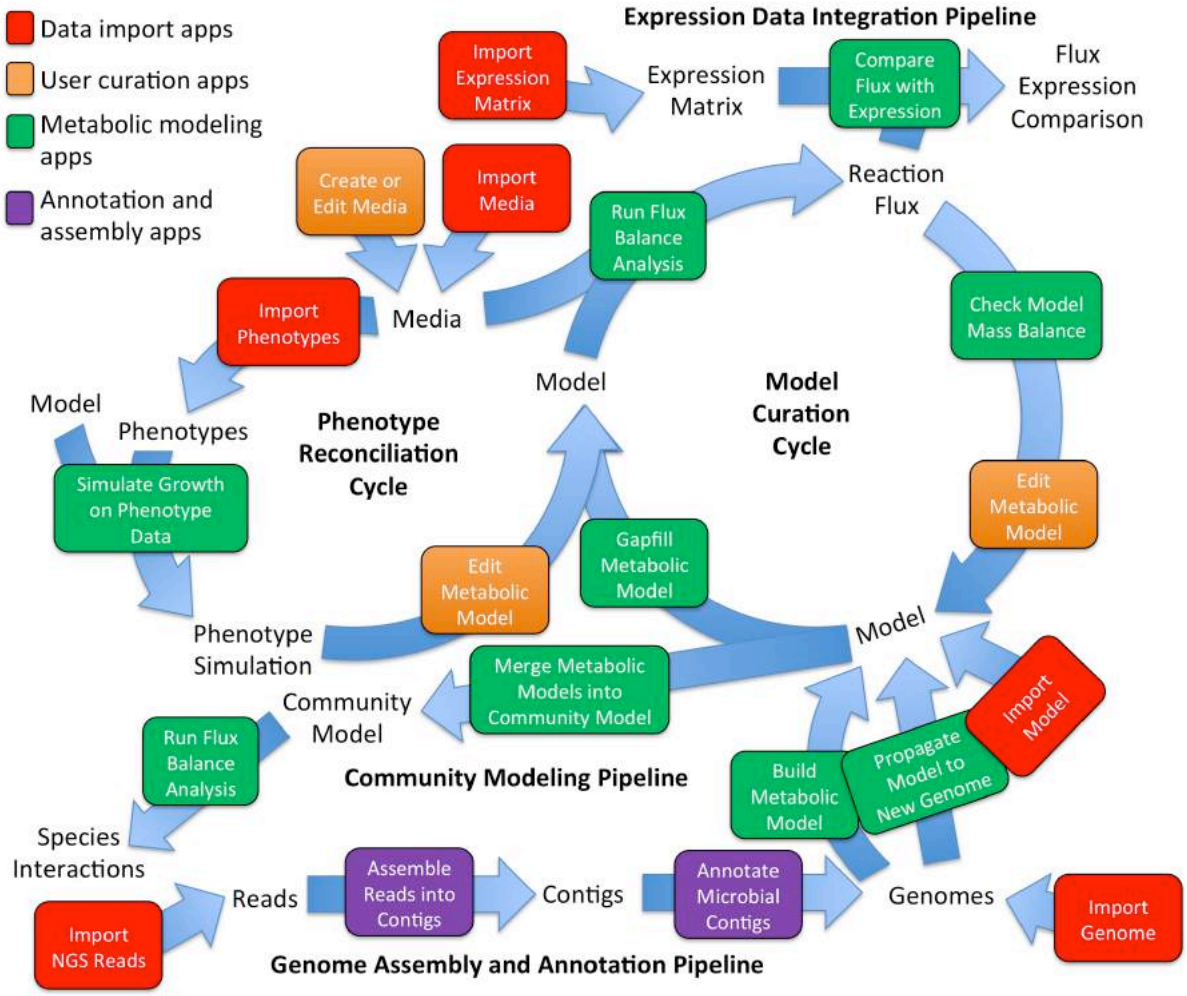


Figure 2. *Workflows in KBase for assembly and annotation of genomes and reconstruction and analysis of metabolic models.*

This report focuses on some of the core workflows available in KBase: assembling genomic sequence reads into contigs, scanning contigs to identify genes, annotating genes with predicted functions, and constructing metabolic models for plant and microbial genomes based on annotated functions. Metabolic models can be used to evaluate an organism's metabolic capability by simulating growth under different conditions to answer important biological questions such as:

- What biochemical pathways are present?
- What are the high flux or essential pathways under a certain growth condition?
- Could the organism grow anaerobically?
- Would it grow under certain minimal media conditions?
- Could the organism be optimized to produce a particular drug molecule or industrially important biofuel?

The flowchart in Figure 2 shows some of the major workflows that presently exist in KBase, including workflows for assembly, annotation and the reconstruction, curation, and analysis of metabolic models.

Assembly and Annotation in KBase

KBase provides pipelines for assembling microbial Next-Generation Sequencing (NGS) short reads and generating annotated genomes from these assemblies (see Figure 2). The starting point for assembly in KBase is a set of single- or paired-end reads. KBase supports the upload of read libraries generated from a variety of sequencing technologies, including Illumina, PacBio CLR, PacBio CSS, IonTorrent, and Oxford Nanopore. Reads files can be uploaded from a stored file or an online site (FTP, HTTP, Dropbox, or Box), or microbial reads can be transferred from the DOE Joint Genome Institute (see http://kbase.us/transfer-jgi-data/).

KBase is beginning to support bulk upload of data, meaning that users can import multiple files simultaneously from their personal computer or remote file servers. This feature is especially beneficial for large-scale analysis of many smaller files or standard analysis of several very large files (e.g., reads). Currently, the pre-beta bulk upload supports reads, genomes, and FASTA files. Numerous other data types (e.g., models, media formulations, phenotype data) are supported by the standard manual upload interface.

To learn about using these assembly and annotation tools see http://kbase.us/assembly-and-annotation/ and start with the Assembling and Annotating Microbial Genomes tutorial or the interactive Narrative tutorial.

Assembly Apps

KBase currently integrates 11 different genome assembly apps. Ten of these are simple wraps of existing assembly algorithms, including A5 [3], A6, IDBA-UD [4], Kiki, MaSuRCA [5], MEGAHIT [6], MiniASM [7], Ray[8], SPAdes [9], and Velvet [10]. These algorithms are described in detail on the KBase website (http://kbase.us/assembly-and-annotation/).

The eleventh app, called *Assemble Contigs from Reads*, runs several different assembly programs and lets users compare the quality of outputs. The app allows users to select from several assembly "recipes" that combine multiple assemblers with other utilities to produce an optimal assembly. The

default option is the Automatic Assembly recipe, which runs three different assemblers (Velvet [10], SPAdes [9], and IDBA-UD [4]), uses BayesHammer [11] for error correction, and chooses an assembly that is suitable for most downstream analyses based on an ARAST quality score. The *Assemble Contigs from Reads* app produces an Assembly data object in addition to an Assembly Report that includes information about the performance of each assembly algorithm that was tested (see Figure 3).

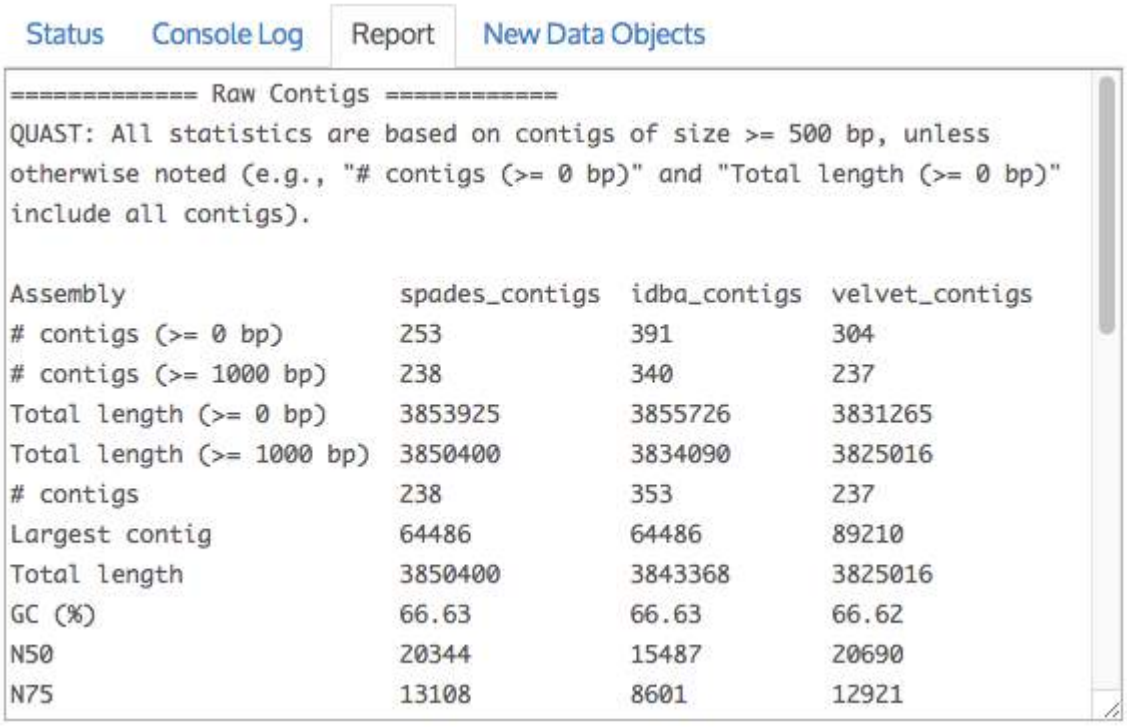


Figure 3. *Example of an Assembly Report generated by the* Assemble Contigs from Reads *app.*

Annotation Apps

Microbial genome annotation in KBase is accomplished using two apps: *Annotate Microbial Contigs* and *Annotate Microbial Genome*. *Annotate Microbial Contigs* takes an Assembly object (assembled contigs) generated by one of the assembly apps as input and applies two different algorithms, Prodigal and Glimmer3, to predict gene locations within the contigs. Next, the gene sequences are passed into the functional annotation pipeline in KBase, which is based on the RAST (Rapid Annotations using Subsystems Technology) toolkit [12]. This annotation pipeline assigns functions from the SEED Subsystems Ontology [13] to genes using a fast kmer-based approach [14]. The *Annotate Microbial Genome* app uses the same functional annotation pipeline as the *Annotate Microbial Contigs* app, but it skips the gene calling step because it is designed to accept a genome with existing gene calls as input.

The KBase annotation pipeline must be applied to genomes before running downstream analysis steps, including metabolic model reconstruction. The annotated Genome object generated by these apps can be explored in a tabular genome viewer that shows summary information about the Genome as well as a list of contigs and the genes that were annotated on each contig (see Figure 4). The genome object can also be exported in GenBank format.

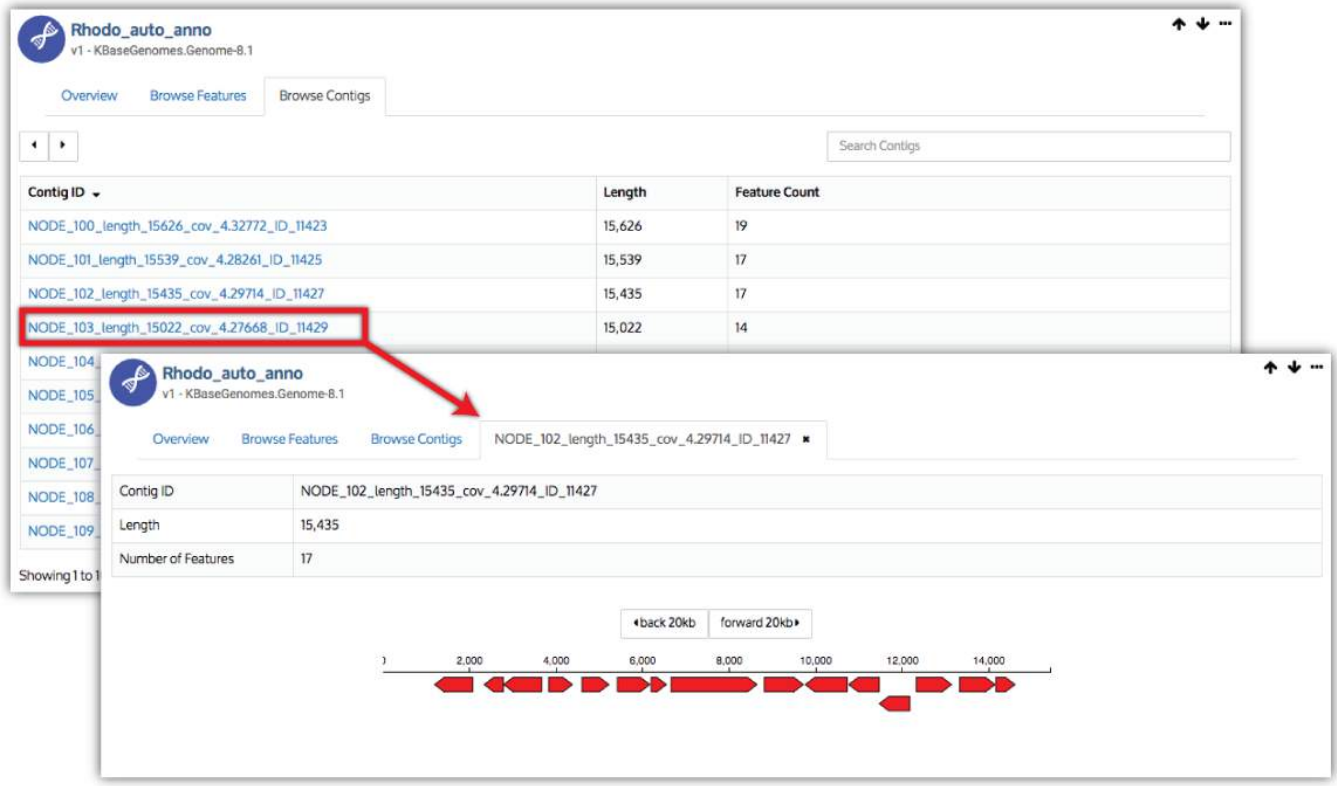


Figure 4. *Example of a genome viewer, showing contigs and genes annotated on a genomic sequence.*

Metabolic Modeling in KBase

KBase has a suite of apps and data that support the reconstruction, optimization, and analysis of metabolic models for microbes and plants (see Figure 2 and http://kbase.us/metabolic-modeling-in-kbase/).

In KBase, genome-scale metabolic models are primarily reconstructed from functional annotations produced by the KBase annotation apps. When a genome is functionally annotated, its metabolic genes are mapped onto biochemical reactions. This information is integrated with data about reaction stoichiometry, subcellular localization, biomass composition, estimation of thermodynamic feasibility (directionality of reactions), and other constraints into a detailed stoichiometric model of metabolism. The "Microbial Metabolic Model Reconstruction and Analysis" Narrative tutorial is an interactive way to see and try out this functionality.

Metabolic modeling in KBase begins with the *Build Metabolic Model* app, which uses the functional annotations generated by the RAST-based genome annotation apps to generate a draft metabolic model [15]. A draft model consists of three parts: (i) a network of metabolic reactions (including both gene-associated reactions and spontaneous reactions); (ii) a set of gene-protein-reaction (GPR) associations that dictate how each reaction activity depends on associated gene activity; and (iii) a biomass composition reaction that defines the small-molecule building blocks that comprise 1 gram of biomass (e.g., amino acids, nucleotides, lipids, cofactors, cell-wall components, and energy). This app produces genome-scale metabolic models by default, but it is possible to build a core model with far fewer reactions by selecting the core template option in the app [16]. Core models are less complex

that genome-scale metabolic models because their biomass composition uses only central-carbon precursor molecules.

Draft metabolic models usually have missing reactions (gaps) due to incomplete or incorrect functional annotations. As a result, these models are unable to produce biomass while simulating growth conditions in which the organism is viable. Gapfilling algorithms can overcome this problem by identifying the minimum number of new reactions that must be added to the model, or existing reactions that must be made reversible, to enable the production of biomass [17, 18]. The *Gapfill Metabolic Model* app can be used to apply these algorithms in KBase.

After building and gapfilling a metabolic model, researchers will typically run Flux Balance Analysis (FBA) to predict reaction fluxes and optimal growth or metabolite production yields. FBA is a constraint-based approach that estimates fluxes through all reactions in the metabolic network by assuming the interior of the cell exists in a quasi-steady state and applying optimization criteria to select a desired flux solution from the otherwise highly under-determined flux space [19]. The most common optimization criterion applied in FBA is the maximization of biomass yield. The *Run Flux Balance Analysis* app in KBase applies the FBA approach to predict the flow of metabolites through the metabolic network of an organism by optimizing a selected cellular objective function, typically biomass maximization. This app allows users to analyze an organism's growth on different substrates and to evaluate the reactions and metabolites that carry fluxes in each growth condition. In addition to optimizing the biomass, users can optimize a certain reaction (e.g., transporter reaction) so that the model maximizes the flux through that reaction.

The *Run Flux Balance Analysis* app requires the user to specify a media formulation in which the growth will be simulated. In KBase, the media contains a list of the chemical compounds that are available for consumption in the flux simulation. KBase currently maintains more than 500 commonly used media conditions. In addition, users are able to build and upload their own custom media formulations.

The *Run Flux Balance Analysis* app also optionally accepts a gene expression profile as input. In this variant of the algorithm, the gene expression values are used to call genes as “on,” “off,” or “unknown.” Genes with very low expression are called “off,” genes with high expression are called “on,” and genes with middling expression values are called “unknown” [20]. The optimization criterion of the FBA is then set to maximize the agreement between the reaction flux and the “on/off” state of the genes associated with each reaction [21, 22]. If all genes associated with a reaction are “off,” then the FBA will attempt to force the flux through the reaction to zero. The power of this approach is that it generates estimates of overall reaction activity throughout the metabolic network that maximize consistency with available expression data, while still allowing some reactions to contradict the expression data.

After running the *Run Flux Balance Analysis* app, two or more models can be compared with the *Compare Models* app. Alternatively, the *Compare FBA Solutions* app can be used to compare flux profiles predicted by FBA to understand how an organism's behavior changes from one condition to the next, or how the behavior of two different organisms differs within a single condition.

Another useful modeling app is *Simulate Growth on Phenotype Data*. This app is also based on the FBA algorithm, but it runs the algorithm in bulk and requires as input a list of growth phenotypes for simulation. In this case, we define a growth phenotype as an indication of growth rate for an organism

on a particular growth condition (media formulation) that may be affected by a particular set of gene knockouts. Biolog phenotype arrays are one of the most common experimental mechanisms for generating growth phenotype data, where the growth of a wild type or mutant strain is evaluated on potentially hundreds of growth conditions [23]. Gene essentiality is another common form of growth phenotype data, where growth of all single-gene knockouts is evaluated in a single growth condition. This app uses FBA to simulate the growth of a model in every specified growth condition and evaluates every specified gene knockout. The user can then assess the quality of the model by comparing predicted growth conditions with observed growth conditions.

Science Performed Using KBase Tools for Assembly, Annotation and Modeling

The first version of the KBase Narrative Interface was launched in February 2015. Since then, more than 1200 users have signed up for KBase accounts, and over 750 have created at least one Narrative. Excluding KBase staff, users have constructed over 3000 Narratives. Most of these Narratives involve the application of the assembly, annotation, and modeling tools to a wide range of scientific use cases. Here we will review some of the publications that cite these tools (see Table 1 and http://kbase.us/publications/). The science covered by these publications falls into three broad categories: (i) annotation and analysis of microbial isolates, (ii) large-scale reconstruction of draft models, and (iii) analysis of microbial communities.

Table 1. Example Narratives demonstrating the use of KBase to annotate microbial genomes, reconstruct microbial metabolic models, and analyze microbial communities (also available at http://kbase.us/kbase-paper/)

Narrative #	Title and URL
1	Comparative Analysis of Phylogenetically Close Genomes and Their Associated Growth Phenotypes (https://narrative.kbase.us/narrative/ws.8773.obj.1)
2	Microbial Comparative Genomics: Reconstruction and Comparison of Core Metabolism Across Microbial Life (https://narrative.kbase.us/narrative/ws.15253.obj.1)
3	Community Modeling Protocol: Multi-Species Model Reconstruction and Analysis (https://narrative.kbase.us/narrative/ws.10824.obj.1)
4	BP1 Meio Community Metabolic Modeling (https://narrative.kbase.us/narrative/ws.13838.obj.1)
5	Electrosynthetic Microbiome: Electrosynthesis Community Model (https://narrative.kbase.us/narrative/ws.15248.obj.1)

Isolate analysis is a popular workflow in KBase since it includes an end-to-end pipeline beginning with raw reads from NGS and culminating in the analysis of a genome-scale metabolic model. In two studies, KBase was applied to annotate and then compare many closely related isolate genomes [24, 25]. In other studies, KBase was used to annotate a genome and then construct and gapfill a draft model, which was subsequently curated and refined with experimental data [26, 27]. In one study of *Klebsiella* KPPR1, KBase was used to simulate and refine a metabolic model to fit available Biolog phenotype data, subsequently identifying inconsistent growth conditions for further experimental study

(Table 1, Narrative 1) [26]. This work demonstrated how the modeling pipeline in KBase can support experimental design. In another analysis that combined these previous workflows, KBase was applied to construct metabolic models for 19 closely related species of *Pseudomonas fluorescens* [28]. The models were then used to study the capacity of these organisms to utilize a variety of different carbon sources. In another study, the gapfilling algorithms in KBase were applied to improve genome annotations by filling gaps in metabolic pathways with reactions for which gene candidates can be identified based on sequence similarity [29]. Finally, in some cases KBase is simply used as a repository for published models, making these models available for easy analysis and download [30].

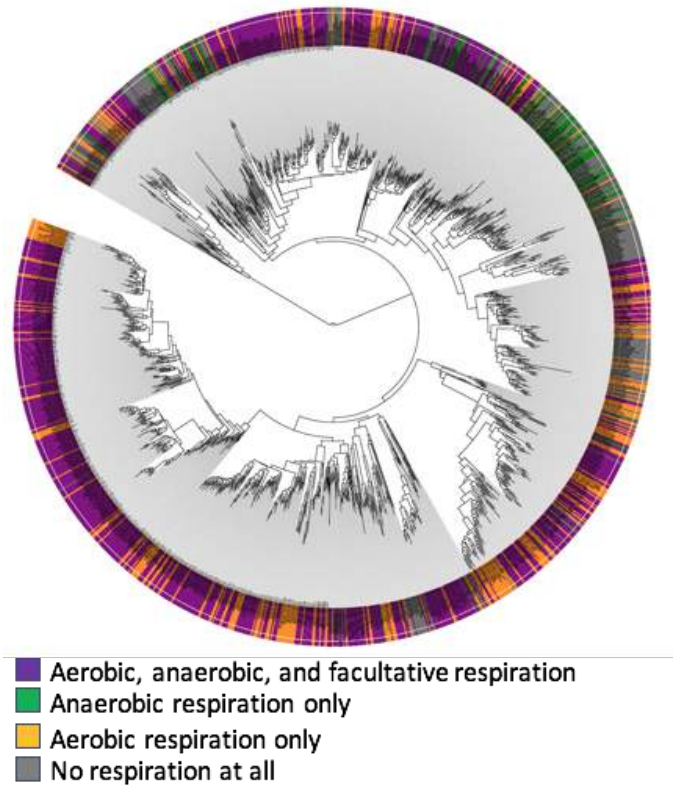
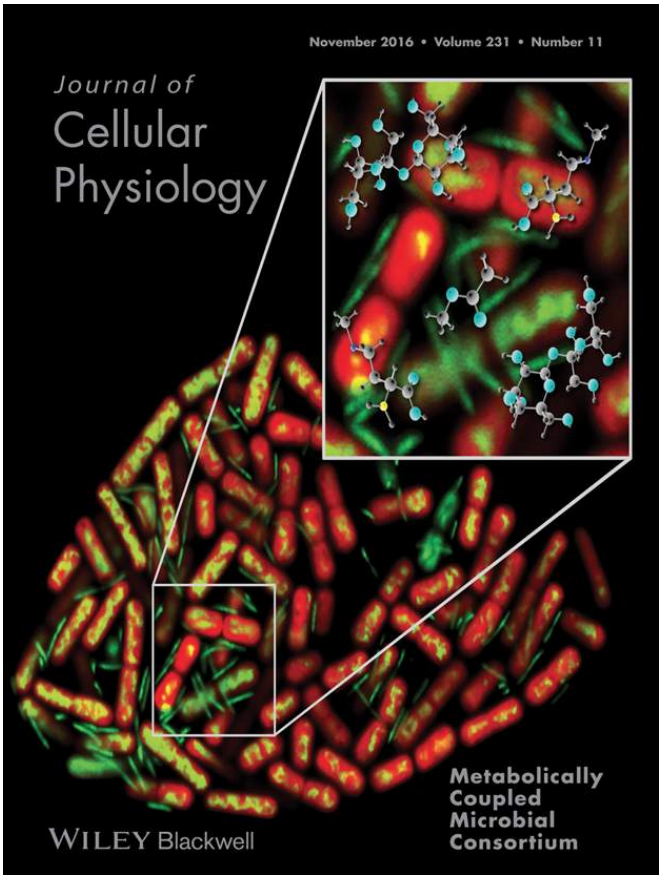


Figure 5 (left). *Distribution of respiration across the microbial tree of life, as described in* Edirisinghe et al. 2016

The KBase annotation and modeling pipeline is able to annotate and draft models for large numbers of genomes. Given the rapidly growing number of available reference genomes, this is an increasingly important capability. In one study, KBase was applied to generate over 8000 draft core metabolic models covering diverse genomes across the microbial tree of life [16] (Table 1, Narrative 2). This analysis revealed patterns in the presence of core metabolic pathways across the microbial phylogenetic tree (see Figure 5), and it provided a comprehensive analysis of energy biosynthesis and fermentation pathways in microbes. In another recent study, KBase was applied to annotate, reconstruct, and gapfill draft models for 773 human gut microbes [31]. The KBase models were subsequently curated and refined, culminating in the largest release of semi-curated models to be published to date. These models significantly enhance our capacity to understand metabolism in the human gut microbiome.

In a growing number of studies, KBase is being used to model the interactions within communities of microbes and plants. The KBase modeling pipeline is distinctive among comparable toolkits for its capability to merge multiple single-genome models together into integrated metabolic models of small microbial communities. This pipeline is demonstrated in three recent publications. One is a book chapter (http://kbase.us/community-modeling/) that explores various microbiome modeling paradigms, predicting potential interactions between the gut microbes *Bacteroides thetaiotaomicron* and *Faecalibacterium prausnitzii* as a case study (Table 1, Narrative 3) [26]. Another analysis examines interactions between an autotrophic carbon-fixing cyanobacteria, *Thermosynechococcus elongatus* BP-1, and a heterotrophic gram-positive species, *Meiothermus ruber* strain A. Model predictions from this analysis were validated by a combination of published growth-condition data and comparison of model-predicted fluxes with metatranscriptome-based expression data (Table 1, Narrative 4) [26]. This work appeared on the cover of the *Journal of Cellular Physiology* in November 2016 (see Figure 6 and



http://kbase.us/predict-interspecies-interactions/). Finally, metagenomic and metatranscriptomic data from an electrosynthetic microbiome were assembled into genomes, models, and flux predictions for three dominant species in the microbiome (Table 1, Narrative 5) [32].

***Figure 6 (left)**. Cover of the* Journal of Cellular Physiology *featuring research done using KBase tools. The image is a confocal micrograph depicting a metabolically coupled microbial consortium composed of the cyanobacterium Thermosynechococcus elongatus (red) supporting a heterotrophic bacterium, Meiothermus ruber (green). Scientists at PNNL and ANL proposed a new approach for microbial community metabolic modeling using this phototroph-heterotroph co-culture as a model system to study how microorganisms living in communities coordinate their metabolisms in response to partnership. Imaged by PNNL scientist William B. Chrisler.*

Conclusion

KBase offers "one-stop shopping" for a growing range of integrated analysis tools and datasets that enable assembly, annotation, and construction and comparison of genome-based metabolic models. One of the most powerful and popular workflows in KBase starts with genomic sequence reads, goes through pipelines for assembly and annotation, and culminates in metabolic models that can be used to evaluate an organism's metabolic capability by simulating growth under different conditions. An increasing number of researchers are using KBase to address research questions in systems biology. A list of publications that cite KBase in their methods can be found at http://kbase.us/publications/.

Q1 Report Authors: Chris Henry, Nomi Harris, Ben Allen
KBase PIs: Adam Arkin, Robert Cottingham, Chris Henry

References

1. Arkin, A.P., et al., *The DOE Systems Biology Knowledgebase (KBase).* bioRxiv, 2016. Preprint first posted online Dec. 22, 2016.
2. Perez, F. and B.E. Granger, *IPython: A system for interactive scientific computing.* Computing in Science & Engineering, 2007. **9**(3): p. 21-29.
3. Coil, D., G. Jospin, and A.E. Darling, *A5-miseq: an updated pipeline to assemble microbial genomes from Illumina MiSeq data.* Bioinformatics, 2015. **31**(4): p. 587-9.
4. Peng, Y., et al. *IDBA–a practical iterative de Bruijn graph de novo assembler*. in *Research in Computational Molecular Biology*. 2010. Springer.
5. Zimin, A.V., et al., *The MaSuRCA genome assembler.* Bioinformatics, 2013. **29**(21): p. 2669-77.
6. Li, D., et al., *MEGAHIT: an ultra-fast single-node solution for large and complex metagenomics assembly via succinct de Bruijn graph.* Bioinformatics, 2015: p. btv033.

7. Li, H., *Minimap and miniasm: fast mapping and de novo assembly for noisy long sequences.* arXiv preprint arXiv:1512.01801, 2015.
8. Boisvert, S., F. Laviolette, and J. Corbeil, *Ray: simultaneous assembly of reads from a mix of high-throughput sequencing technologies.* J Comput Biol, 2010. **17**(11): p. 1519-33.
9. Bankevich, A., et al., *SPAdes: a new genome assembly algorithm and its applications to single-cell sequencing.* Journal of Computational Biology, 2012. **19**(5): p. 455-477.
10. Zerbino, D.R. and E. Birney, *Velvet: algorithms for de novo short read assembly using de Bruijn graphs.* Genome research, 2008. **18**(5): p. 821-829.
11. Nikolenko, S.I., A.I. Korobeynikov, and M.A. Alekseyev, *BayesHammer: Bayesian clustering for error correction in single-cell sequencing.* BMC genomics, 2013. **14**(1): p. 1.
12. Brettin, T., et al., *RASTtk: a modular and extensible implementation of the RAST algorithm for building custom annotation pipelines and annotating batches of genomes.* Sci Rep, 2015. **5**: p. 8365.
13. Overbeek, R., et al., *The subsystems approach to genome annotation and its use in the project to annotate 1000 genomes.* Nucleic Acids Research, 2005. **33**(17): p. 5691-5702.
14. Edwards, R.A., et al., *Real time metagenomics: using k-mers to annotate metagenomes.* Bioinformatics, 2012. **28**(24): p. 3316-7.
15. Henry, C.S., et al., *High-throughput generation, optimization, and analysis of genome-scale metabolic models.* Nature Biotechnology, 2010. **Nbt.1672**: p. 1-6.
16. Edirisinghe, J.N., et al., *Modeling central metabolism and energy biosynthesis across microbial life.* BMC Genomics, 2016. **17**: p. 568.
17. Latendresse, M., *Efficiently gap-filling reaction networks.* BMC Bioinformatics, 2014. **15**: p. 225.
18. Dreyfuss, J.M., et al., *Reconstruction and validation of a genome-scale metabolic model for the filamentous fungus Neurospora crassa using FARM.* PLoS Comput Biol, 2013. **9**(7): p. e1003126.
19. Orth, J.D., I. Thiele, and B.O. Palsson, *What is flux balance analysis?* Nat Biotechnol, 2010. **28**(3): p. 245-8.
20. Faria, J.P., et al., *Computing and Applying Atomic Regulons to Understand Gene Expression and Regulation.* Front Microbiol, 2016. **7**: p. 1819.
21. Seaver, S.M., et al., *Improved evidence-based genome-scale metabolic models for maize leaf, embryo, and endosperm.* Front Plant Sci, 2015. **6**: p. 142.
22. Becker, S.A. and B.O. Palsson, *Context-specific metabolic networks are consistent with experiments.* PLoS Comput Biol, 2008. **4**(5): p. e1000082.
23. Mackie, A.M., et al., *Biolog Phenotype Microarrays for phenotypic characterization of microbial cells.* Methods Mol Biol, 2014. **1096**: p. 123-30.
24. Ormerod, K.L., et al., *Genomic characterization of the uncultured Bacteroidales family S24-7 inhabiting the guts of homeothermic animals.* Microbiome, 2016. **4**(1): p. 36.
25. Faria, J.P., et al., *Enabling comparative modeling of closely related genomes: example genus Brucella.* 3 Biotech, 2014: p. 1-5.
26. Henry, C.S., et al., *Generation and validation of the iKp1289 metabolic model for Klebsiella pneumoniae KPPR1.* Journal of Infectious Disease, 2016. In press.
27. diCenzo, G.C., et al., *Metabolic modelling reveals the specialization of secondary replicons for niche adaptation in Sinorhizobium meliloti.* Nat Commun, 2016. **7**: p. 12219.
28. Timm, C.M., et al., *Metabolic functions of Pseudomonas fluorescens strains from Populus deltoides depend on rhizosphere or endosphere isolation compartment.* Front Microbiol, 2015. **6**: p. 1118.
29. Benedict, M.N., et al., *Likelihood-based gene annotations for gap filling and quality assessment in genome-scale metabolic models.* PLoS Comput Biol, 2014. **10**(10): p. e1003882.
30. Thompson, R.A., et al., *Exploring complex cellular phenotypes and model-guided strain design with a novel genome-scale metabolic model of Clostridium thermocellum DSM 1313 implementing an adjustable cellulosome.* Biotechnol Biofuels, 2016. **9**(1): p. 194.
31. Magnusdottir, S., et al., *Generation of genome-scale metabolic reconstructions for 773 members of the human gut microbiota.* Nat Biotechnol, 2016.
32. Marshall, C., et al., *Electron transfer and carbon metabolism in an electrosynthetic microbial community.* mSystems, 2016. Submitted.