`kmindex` and `ORA`: indexing and real-time user-friendly queries in terabyte-sized complex genomic datasets

Téo Lemane[1,*], Nolan Lezzoche[2], Julien Lecubin[3], Eric Pelletier[4,5], Magali Lescot[2,5], Rayan Chikhi[6], and Pierre Peterlongo[1,*]

[1]Univ. Rennes, Inria, CNRS, IRISA - UMR 6074, Rennes, F-35000 France
[2]Aix-Marseille Université, Université de Toulon, IRD, CNRS, Mediterranean Institute of Oceanography (MIO), UM 110, Marseille, France
[3]SIP, OSU PYTHEAS, Marseille, France
[4]Génomique Métabolique, Genoscope, Institut de Biologie François Jacob, CEA, CNRS, Univ. Evry, Université Paris-Saclay, Evry, France
[5]Research Federation for the Study of Global Ocean Systems Ecology and Evolution, FR2022/Tara Oceans GO-SEE, CNRS, Paris, France
[6]Institut Pasteur, Université Paris Cité, G5 Sequence Bioinformatics, Paris, F-75015, France
[*]Corresponding authors: teo.lemane@genoscope.cns.fr, pierre.peterlongo@inria.fr

Abstract

Public sequencing databases contain vast amounts of biological information, yet they are largely underutilized as one cannot efficiently search them for any sequence(s) of interest. We present `kmindex`, an innovative approach that can index thousands of highly complex metagenomes and perform sequence searches in a fraction of a second. The index construction is an order of magnitude faster than previous methods, while search times are two orders of magnitude faster. With negligible false positive rates below 0.01%, `kmindex` outperforms the precision of existing approaches by four orders of magnitude. We demonstrate the scalability of `kmindex` by successfully indexing 1,393 complex marine seawater metagenome samples from the *Tara* Oceans project. Additionally, we introduce the publicly accessible web server "Ocean Read Atlas" (`ORA`) at `https://ocean-read-atlas.mio.osupytheas.fr/`, which enables real-time queries on the *Tara* Oceans dataset. The open-source `kmindex` software is available at `https://github.com/tlemane/kmindex`.

Public genomic datasets are growing at an exponential rate [8]. They contain treasures of genomic information that enable groundbreaking discoveries in fundamental domains such as agronomy, ecology, and health [10, 19]. Unfortunately, despite their public availability in repositories such as the Sequence Read Archive [14], these resources, measured in petabytes, are rarely ever reused globally because they cannot be searched. Recent years have seen many methodological developments towards sequencing data search engines (see [5] and [16] for a review).

Current methods for searching genomic sequencing data look for k-mers (words of fixed length k, with k usually in $[20; 50]$) shared between a query sequence and each sample present in a reference database. The central operation is thus to determine, for each k-mer, in which indexed sample(s) it occurs.

In this work, we focus on the challenge of indexing and querying large and complex metagenomic datasets. Given the data size, i.e. thousands of samples totalling tens of terabytes of compressed data, and its complexity, i.e. thousands of billions of distinct k-mers, the computational challenge is immense. Once k-mers are extracted from raw data and filtered, a data structure is built to associate each k-mer to the sample(s) in which it occurs.

Techniques for association k-mers to samples can be divided into three categories: 1. sketching approaches that heavily subsample k-mers; 2. exact data structures storing all k-mers; 3. approximate membership data structures e.g. Bloom filters. Sketching approaches such as sourmash [20], or Needle [9] typically suffer from high false negative rates when short sequences are queried, and are thus out of the scope of this work. Methods based on exact representations (e.g. `MetaGraph` [13], `BiFrost` [12], and `ggcat` [6]) suffer from low scalability, as highlighted by our results. We are thus left with methods based on Bloom filters (BFs) [4], such as `COBS` [3], `SBT` [21], later improved by `HowDeSBT` [11], and more recently by `MetaProFi` [22] able to index billions of k-mers using only a few dozen of gigabytes of space.

When indexing large and complex metagenomic datasets, existing tools face significant limitations in either 1. disk usage; 2. memory usage; 3. computation time (either during indexing and/or query); 4. false positive rate; and 5. false negative rate. Overcoming simultaneously all these limitations makes the design of an efficient data indexing strategy particularly challenging. We present `kmindex`, a new tool that performs indexing and queries using orders of magnitudes less resources than previous approaches. Also, `kmindex` provides results with no false negative calls and with negligible false positive (FP) rates, approximately four orders of magnitude smaller than those obtained by other tools. `kmindex` is primarily designed for indexing complex sequencing samples. Due to engineering choices, it is currently not suited for indexing large collections of genomes (i.e. hundreds of thousands of samples).

To showcase the features of `kmindex` on a dataset of high biological interest, we introduce a web server named "Ocean Read Atlas" (`ORA`) available at this URL: `https://ocean-read-atlas.mio.osupytheas.fr/`. `ORA` allows to search one or several sequences across all of *Tara* Oceans metagenomic raw sequencing data [23]. It enables the visualization of the results on a geographic map and their correlation with each of the 56 environmental variables collected during the circumnavigation campaign. The `ORA` server enables to perform instant searches on a large and complex dataset, providing new perspectives on the deep exploitation of *Tara* Oceans resources.

Results

Comparative results indexing 50 metagenomic seawater samples

We evaluated the performances of `kmindex` together with eight state-of-the-art k-mer indexers: `themisto` [2]; `ggcat` [7]; `HIBF` [18]; `PAC` [17]; `MetaProFi` [22]; `MetaGraph` [13]; `Bifrost` [12]; and `COBS` [3]. The dataset for this benchmark is composed of metagenomic seawater sequencing data from 50 *Tara* Oceans samples, of 1.4TB of gzipped fastq files. It contains approximately 1,420 billion k-mers. Among them, approximately 394 billion are distinct, and 132 billion occur twice or more. The exhaustive list of tool versions and commands used are proposed in a companion website `https://github.com/pierrepeterlongo/kmindex_benchmarks`, that also reports the False Positive computation protocols and a detailed description of the dataset considered for this benchmark.

The benchmarking setup is described in Supplementary Materials. The extensive results of all the following claims are proposed in the Supplementary Materials.

`kmindex` has superior index construction performance. Among the nine tested tools, only `MetaProFi`, `COBS`, and `kmindex` completed the index creation phase and were able to perform queries correctly. As shown in Table 1, building an index with `kmindex` is an order of magnitude faster than `MetaProFi` and

	Build index				Query time		FP rate (%)	
	Time	RAM GB	Disk GB	Index size GB	Nb. queried reads 1	Nb. queried reads 10 million	Average	Max
`MetaProFi`	30h15	278	5,684	226	12s72	1h29	11.18	21.55
`COBS`	26h30	278	5,684	184	1s51	15h56	13.29	24.60
`kmindex`	**2h56**	**107**	**878**	**164**	**0s06**	**4m21s**	**0.006**	**0.18**

Table 1: Overview of index construction and read query performance of `kmindex` compared to `MetaProFi` and `COBS`, on 50 *Tara* Ocean samples. These are the only tools that succeeded in building an index and perform queries. "RAM" and "Disk" columns provide the peak usage during the building process. `COBS` and `MetaProFi` RAM and disk peaks are identical as they correspond to the same k-mer counting and filtration step. Queries are composed of one read and 10 million reads uniformly sampled from the 50 *Tara* Oceans datasets. All executions were performed on a cold cache. Extended results are proposed in the Supplementary Materials.

`COBS`, and uses 2.6x less memory and 6.5x less disk. The final index sizes are all within the same magnitude range, with the smallest one produced by `kmindex`. The `kmindex` construction took less than 3 hours, a peak RAM of 107GB, and a peak disk usage of 878GB.

`kmindex` enables real-time queries As shown in Table 1, at query time, `kmindex` outperforms `MetaProFi` and `COBS`, both in terms of computation time (and memory resources, see Supp Matt). `kmindex` is between 20 and 200 times faster than `MetaProFi` and `COBS` for querying one read or millions of reads. `kmindex` is capable of performing millions of queries in a matter of minutes while allowing real-time resolution for small queries. This opens the doors to analyzing complete read sets as queries, and the deployment of real-time query servers as presented in the next section. Of note, `kmindex` also offers a "fast-mode", presented in the Supplementary Materials, that uses more RAM to achieve even faster queries.

`kmindex` allows highly accurate queries. The `kmindex`, `MetaProFi`, and `COBS` scalability is achieved thanks to the usage of Bloom Filters that generate False Positive (FP) calls at query time. FP rate analyses, summarized Table 1, show that for a similar index size, `MetaProFi` and `COBS` present sensible false positive hits, on average 11.18% and 13.29% respectively over the 50 answers (one per indexed sample). In contrast, the `kmindex` FP rate is negligible (below 10^{-2}% on average).

Indexing 1,393 *Tara* Oceans samples in the Ocean Read Atlas web server

Thanks to these novel possibilities offered by `kmindex`, we built and made available a public web interface able to perform queries on a dataset composed of 1,393 samples (distinct locations and distinct fraction sizes) of the *Tara* Oceans project [23] representing 36.7 TB of raw fastq.gz files. A user can query sequences, determining their similarity with the 1,393 indexed samples. A world map depicts the resulting biogeography, as well as the environmental parameters associated with the sequences.

Note that for reasons of robustness and continuity of service, the index is deployed on a networked and redundant filesystem with lower performances compared to the benchmark environment, although suitable for this type of service. Details about indexed read sets, and more information about the server architecture and setup are provided in Supplementary Materials.

The resulting web server named "Ocean Read Atlas" (`ORA`), whose representation is provided in Figure 1, extends the "Ocean Gene Atlas" server (OGA) [25, 24] that supports queries to assembled genes from *Tara* Oceans [23] and Malaspina [1]. We believe this server will be of great importance to the *Tara* Oceans consortium as a whole, and more broadly to anybody interested in marine genetic data.

`kmindex` provides a high level of usability

Index dynamicity. `kmindex` enables to add new samples to an index. A novel and independent index can be *registered* with a previous one. At query time, each registered index is queried independently. This offers the possibility to query only a subset of the registered indexes. This is well adapted when indexing samples with distinct characteristics. Alternatively, users can extend an existing index, and the parameters of the previous index (such as the *ad-hoc* hash function or the BFs sizes) are automatically

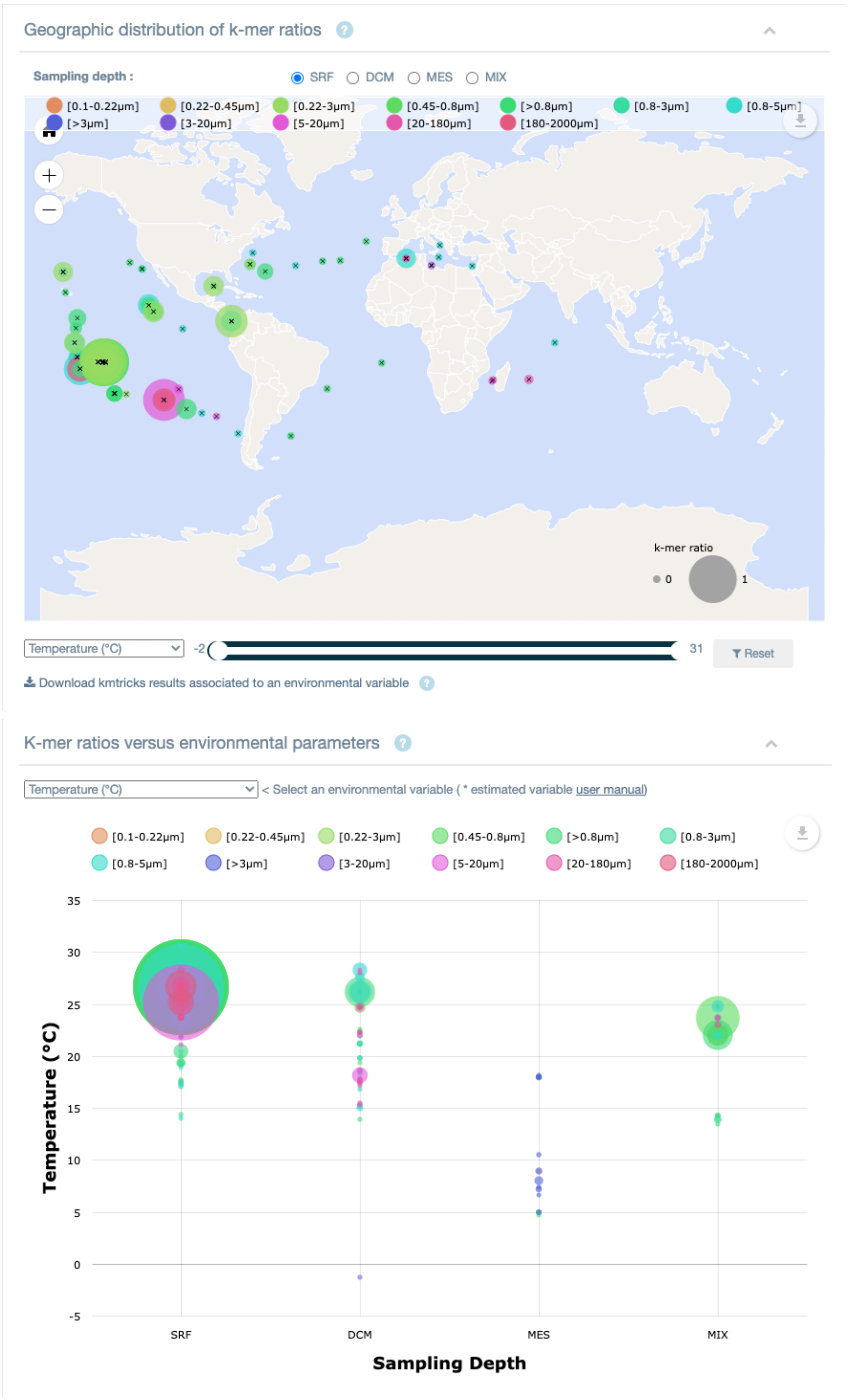

Figure 1: Screenshot of the "Ocean Read Atlas" result interface. Top: the biogeography distribution of the queried sequence is shown among all data samples. The size of the point depicts the similarity of the queried sequences with the corresponding sample. Bottom: a bubble plot representing the correlation between the query presence and the environmental variables of the samples in which it occurs.

reused. This second choice is less flexible but provides better performances at query time (see results presented in Supplementary Materials).

Versatile k-mer filtration. `kmindex` enables the filtration of erroneous k-mers, not only relying on their abundance in a dataset but also on their co-abundances in all indexed datasets. This enables to "rescue" low-abundance k-mers that would have otherwise been removed. To the best of our knowledge, no other indexing tool can integrate this feature. This feature is inherited from the `kmtricks` [15] algorithm.

Variable query resolution. `kmindex` query results can be provided with various degrees of precision. For each indexed sample, users can access the average similarity of queried sequences or a similarity value per queried sequence. Finally, `kmindex` can provide the distribution of hits, enabling to highlight some regions of interest among the queried sequences.

High accessibility. `kmindex` is well documented and simple to install. Queries can be performed via a CLI, via an API, or as an HTTP server.

Methods

We briefly sum up here the `kmindex` method, while the complete description is provided in the Supplementary Materials. Conceptually, the presence of each indexed k-mer is stored in one BF per input read set. The BFs construction relies on `kmtricks` [15] which allows to filter erroneous k-mers and to efficiently build a partitioned matrix of BFs. Each partition indexes a subset of k-mers corresponding to a specific set of minimizers. In practice, with `kmindex`, matrices are *inverted* to limit cache misses during the query process, i.e. each row is a bit vector representing the presence/absence of a k-mer in each indexed sample. At query time, k-mers from queried sequences are grouped into batches, and, avoiding cache misses, BFs are queried to determine the presence or absence of each k-mer in each input dataset.

Conclusion

We propose `kmindex`, a tool for creating k-mer indexes from terabyte-sized raw sequencing datasets. It is the only tool able to index highly complex data such as thousands of seawater metagenomic samples, and to provide instant query answers, with a non-zero but negligible false positive rate, in average below 0.01% in our tests. By its performance and its usage simplicity, `kmindex` makes indexing k-mers from large and complex genomic projects practically possible for the first time.

We believe that `kmindex` opens up a new channel for leveraging genetic data, removing the obstacles that often isolate studies from each other. The "Ocean Read Atlas" illustrates this advance, by providing a highly usable tool to fully utilize the wealth of data generated by the *Tara* Oceans project.

Data Availability

A list of publicly available data used in this work is proposed in the `https://github.com/pierrepeterlongo/kmindex_benchmarks` repository.

Funding

The work was funded by ANR SeqDigger (ANR-19-CE45-0008), the IPL Inria Neuromarkers, and received some support from the French government under the France 2030 investment plan, as part of the Initiative d'Excellence d'Aix-Marseille Université - A*MIDEX - Institute of Ocean Sciences (AMX-19-IET-016). This work is part of the ALPACA project that has received funding from the European Union's Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grants agreements No 956229 and 872539 (PANGAIA). R.C. was supported by ANR Full-RNA, Inception and PRAIRIE grants (ANR-22-CE45-0007, PIA/ANR16-CONV-0005, ANR-19-P3IA-0001).

Acknowledgements

We acknowledge the GenOuest core facility (`https://www.genouest.org`) and the TGCC (`https://www-hpc.cea.fr/index-en.html`) for providing the computing infrastructure, as well as France Génomique for funding of the TGCC computing resources used to process data used in this article. The authors thank Jean-Marc Aury for his help regarding the usage of the *Tara* Oceans data sets. *Tara* Oceans (which includes both the *Tara* Oceans and *Tara* Oceans Polar Circle expeditions) would not exist without the leadership of the *Tara* Ocean Foundation and the continuous support of *Tara* Oceans consortium members. The authors also thank Kahles Andre and Mustafa Harun for their help regarding the usage of `MetaGraph`, Andrea Cracco and Alexandru Tomescu for their help using `ggcat`, and Camille Marchet and Antoine Limasset for their support using `PAC`. The web server is hosted by the OSU Pythéas cluster with the help of Cyrille Blanpain and SIP members. Adrien Malgoyre from SIP is thanked for the development of the OSU Pythéas GitLab.

References

[1] S.G Acinas, P Sánchez, G Salazar, F.M. Cornejo-Castillo, M. Sebastián, R. Logares, M Royo-Llonch, L Paoli, S. Sunagawa, P. Hingamp, and et al. Deep ocean metagenomes provide insight into the metabolic architecture of bathypelagic microbial communities. *Communications Biology*, 4(604):1–15, 2022.

[2] Jarno N Alanko, Jaakko Vuohtoniemi, Tommi Mäklin, and Simon J Puglisi. Themisto: a scalable colored k-mer index for sensitive pseudoalignment against hundreds of thousands of bacterial genomes. *bioRxiv*, pages 2023–02, 2023.

[3] Timo Bingmann, Phelim Bradley, Florian Gauger, and Zamin Iqbal. Cobs: a compact bit-sliced signature index. In *String Processing and Information Retrieval: 26th International Symposium, SPIRE 2019, Segovia, Spain, October 7–9, 2019, Proceedings 26*, pages 285–303. Springer, 2019.

[4] Burton H Bloom. Space/time trade-offs in hash coding with allowable errors. *Communications of the ACM*, 13(7):422–426, 1970.

[5] Rayan Chikhi, Jan Holub, and Paul Medvedev. Data structures to represent a set of k-long dna sequences. *ACM Computing Surveys (CSUR)*, 54(1):1–22, 2021.

[6] Andrea Cracco and Alexandru I Tomescu. Extremely-fast construction and querying of compacted and colored de bruijn graphs with ggcat. *bioRxiv*, pages 2022–10, 2022.

[7] Andrea Cracco and Alexandru I Tomescu. Extremely fast construction and querying of compacted and colored de bruijn graphs with ggcat. *Genome Research*, pages gr–277615, 2023.

[8] Carla Cummins, Alisha Ahamed, Raheela Aslam, Josephine Burgin, Rajkumar Devraj, Ossama Edbali, Dipayan Gupta, Peter W Harrison, Muhammad Haseeb, Sam Holt, et al. The european nucleotide archive in 2021. *Nucleic Acids Research*, 50(D1):D106–D110, 2022.

[9] Mitra Darvish, Enrico Seiler, Svenja Mehringer, René Rahn, and Knut Reinert. Needle: a fast and space-efficient prefilter for estimating the quantification of very large collections of expression experiments. *Bioinformatics*, 38(17):4100–4108, 07 2022.

[10] Robert C Edgar, Jeff Taylor, Victor Lin, Tomer Altman, Pierre Barbera, Dmitry Meleshko, Dan Lohr, Gherman Novakovsky, Benjamin Buchfink, Basem Al-Shayeb, et al. Petabase-scale sequence alignment catalyses viral discovery. *Nature*, 602(7895):142–147, 2022.

[11] Robert S Harris and Paul Medvedev. Improved representation of sequence bloom trees. *Bioinformatics*, 36(3):721–727, 2020.

[12] Guillaume Holley and Páll Melsted. Bifrost: highly parallel construction and indexing of colored and compacted de bruijn graphs. *Genome biology*, 21(1):1–20, 2020.

[13] Mikhail Karasikov, Harun Mustafa, Daniel Danciu, Marc Zimmermann, Christopher Barber, Gunnar Rätsch, and André Kahles. Metagraph: Indexing and analysing nucleotide archives at petabase-scale. *BioRxiv*, 2020.

[14] Kenneth Katz, Oleg Shutov, Richard Lapoint, Michael Kimelman, J Rodney Brister, and Christopher O'Sullivan. The sequence read archive: a decade more of explosive growth. *Nucleic acids research*, 50(D1):D387–D390, 2022.

[15] Téo Lemane, Paul Medvedev, Rayan Chikhi, and Pierre Peterlongo. kmtricks: Efficient and flexible construction of bloom filters for large sequencing data collections. *Bioinformatics Advances*, 2022.

[16] Camille Marchet, Christina Boucher, Simon J Puglisi, Paul Medvedev, Mikaël Salson, and Rayan Chikhi. Data structures based on k-mers for querying large collections of sequencing data sets. *Genome Research*, 31(1):1–12, 2021.

[17] Camille Marchet and Antoine Limasset. Scalable sequence database search using Partitioned Aggregated Bloom Comb-Trees. In *Recomb 2022- 26th Annual International Conference on Research in Computational Molecular Biology*, La jolla, United States, May 2022.

[18] Svenja Mehringer, Enrico Seiler, Felix Droop, Mitra Darvish, René Rahn, Martin Vingron, and Knut Reinert. Hierarchical interleaved bloom filter: enabling ultrafast, approximate sequence queries. *Genome Biology*, 24(1):1–25, 2023.

[19] Lucas Paoli, Hans-Joachim Ruscheweyh, Clarissa C Forneris, Florian Hubrich, Satria Kautsar, Agneya Bhushan, Alessandro Lotti, Quentin Clayssen, Guillem Salazar, Alessio Milanese, et al. Biosynthetic potential of the global ocean microbiome. *Nature*, 607(7917):111–118, 2022.

[20] N. Tessa Pierce, Luiz Irber, Taylor Reiter, Phillip Brooks, and C. Titus Brown. Large-scale sequence comparisons with sourmash. *F1000Research*, 8:1006, July 2019.

[21] Brad Solomon and Carl Kingsford. Improved search of large transcriptomic sequencing databases using split sequence bloom trees. *Journal of Computational Biology*, 25(7):755–765, 2018.

[22] Sanjay K Srikakulam, Sebastian Keller, Fawaz Dabbaghie, Robert Bals, and Olga V Kalinina. Metaprofi: an ultrafast chunked bloom filter for storing and querying protein and nucleotide sequence data for accurate identification of functionally relevant genetic variants. *Bioinformatics*, 39(3):btad101, 2023.

[23] Shinichi Sunagawa, Silvia G Acinas, Peer Bork, Chris Bowler, Tara Oceans Coordinators, Damien Eveillard, Gabriel Gorsky, Lionel Guidi, Daniele Iudicone, Eric Karsenti, Fabien Lombard, Hiroyuki Ogata, Stephane Pesant, Matthew B Sullivan, Patrick Wincker, and Colomban de Vargas. Tara oceans: towards global ocean ecosystems biology. *Nat Rev Microbiol*, 18(8):428–445, 2020.

[24] Caroline Vernette, Julien Lecubin, Pablo Sánchez, Shinichi Sunagawa, Tom O Delmont, Silvia G Acinas, Eric Pelletier, Pascal Hingamp, and Magali Lescot. The ocean gene atlas v2. 0: online exploration of the biogeography and phylogeny of plankton genes. *Nucleic Acids Research*, 50(W1):W516–W526, 2022.

[25] Emilie Villar, Thomas Vannier, Caroline Vernette, Magali Lescot, Miguelangel Cuenca, Aurélien Alexandre, Paul Bachelerie, Thomas Rosnet, Eric Pelletier, Shinichi Sunagawa, et al. The ocean gene atlas: exploring the biogeography of plankton genes online. *Nucleic Acids Research*, 46(W1):W289–W295, 2018.