SPIRE 2023

Indexing Large Metagenomic Projects

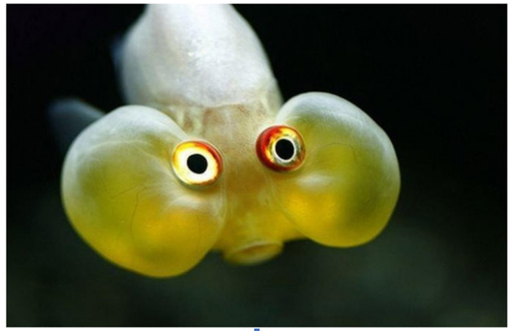
Application to the Tara Oceans Datasets

©Antoine Doré Illustration

Téo Lemane, Lucas Robidou, Rayan Chikhi, Pierre Peterlongo

IRISA Inria GenOuest SeqDigger mio Institut Méditerranéen d'Océanologie GenScale

Sequencing a genome



```
>read1
ACCGGCAGCAGTCTCAGCATGACTCGATCGCGAGCAT
>read2
CGGCATCGTGGATCGCAGGAGTACGATCGTCAGAGTA
…
…
…
…
>read500,000000
CGTACGCAGGCAGCGGCATCAATGACTGTGTGTGTCA
```

One "run" (12h):
300GB raw data

Three runs:
1TB. One laptop

Sequencing a metagenome

100000 species

```
>read1
ACCGGCAGCAGTCTCAGCATGACTCGATCGCGAGCAT
>read2
CGGCATCGTGGATCGCAGGAGTACGATCGTCAGAGTA
…
…
…
…
>read500,000000
CGTACGCAGGCAGCGGCATCAATGACTGTGTGTGTCA
```

One “run” (12h):
  300GB raw data

Three runs:
  1TB. One laptop

Which data?

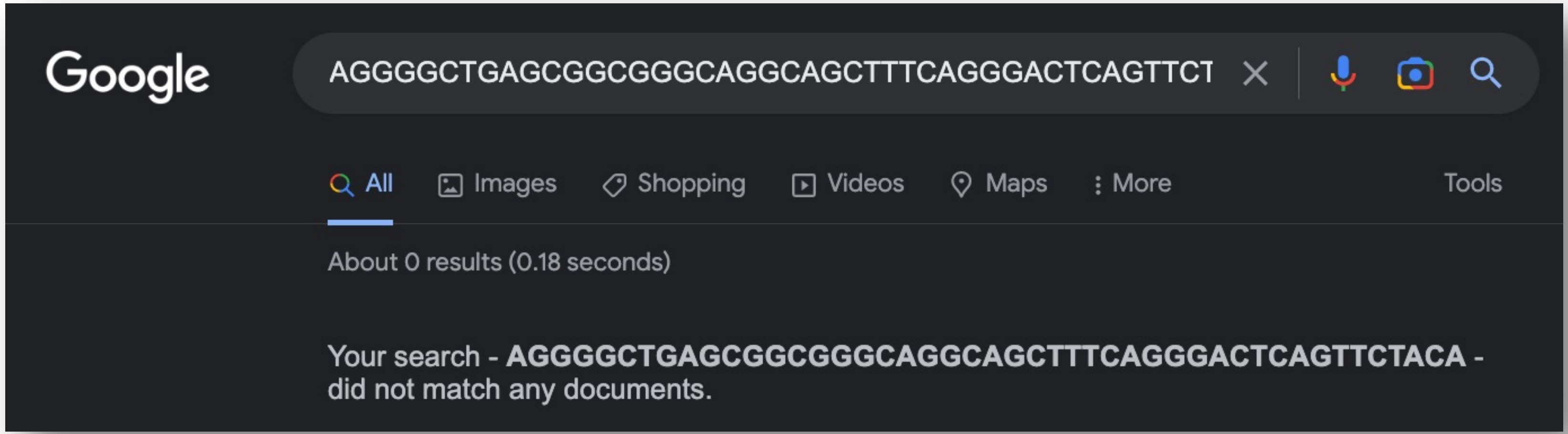
Raw sequences

- **Fragmented** data
- Error-prone (1% to 10% error rate)
  - Important **background noise**
- Heterogenous
  - Quality and quantity
- **Volumes**:
  - hundreds millions fragments / experiment
  - Millions of experiments
- **Archived**

ENA European Nucleotide Archive

51 petabytes

doubles: <2 years



AGGGGCTGAGCGGCGGGCAGGCAGCTTTCAGGGACTCAGTTCT

About 0 results (0.18 seconds)

Your search - **AGGGGCTGAGCGGCGGGCAGGCAGCTTTCAGGGACTCAGTTCTACA** - did not match any documents.

Objectives

At the price of
- Approximate answers (FP, overestimations)
- Need fast disk (local SSD preferentially)

Index

- Genomic datasets:
  - Large
    - > hundreds, thousand samples
    - TB to PB sized
  - **Complex**
    - metagenomes, metatranscriptomes,
    - high variability (sea water, soil, …)
- With:
  - Low RAM usage
    - (max 100 GB)
  - Dynamicity
    - Able to add new samples to the index
  - Fast
    - Indexing hundreds of samples in a few hours

Query

- Short (reads) or long (genomes) sequences
- One sequence (google-like)
  - Real time (milliseconds)
  - No RAM

Or

- Many sequences (read set)
  - Fast (~hours)
  - RAM limited (max 100 GB)
- With or without abundance

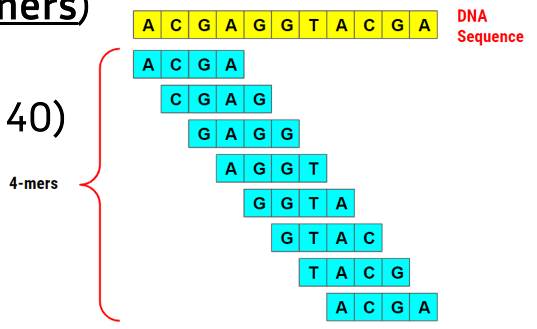
Genomic research engine: conceptual view

kmers

Words

- **No word in DNA**
- Split to subsequences of fixed length *k* (called **kmers**)

(20 < *k* < 40)



- Thousand billions distinct kmers
  - *(google indexes millions)*

Compare sequences

- Sequence similarity ~ shared kmers count

```
ACGAGGTACGA        ACGAGTTACGA
ACGA               ACGA
 CGAG               CGAG
  GAGG               GAGT
   AGGT               AGTT
    GGTA               GTTA
     GTAC               TTAC
      TACG               TACG
       ACGA               ACGA
```

- 4 over 8 kmers shared

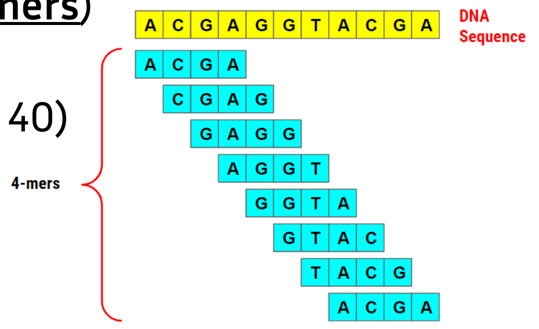
Genomic research engine: conceptual view

kmers

Words

- **<u>No word in DNA</u>**
- Split to subsequences of fixed length *k* (called **<u>kmers</u>**)

(20 < *k* < 40)



- Thousand billions distinct kmers
  - *(google indexes millions)*

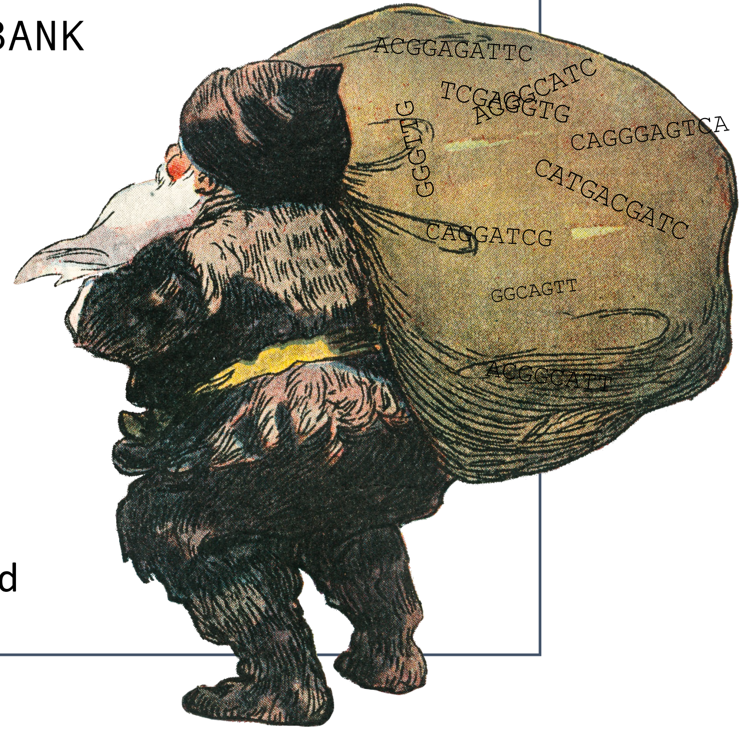
Query vs Bank

- Sequence similarity ~ shared kmers count

ACGAGGTACGA BANK

ACGA
CGAG
GAGG
AGGT
GGTA
GTAC
TACG
ACGA

- 6 over 8 kmers shared



Genomic research engine: conceptual view

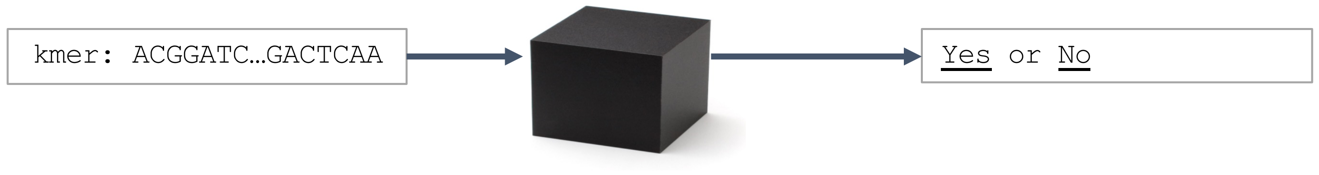
index

Set representation

- A bank (genome, reads, …) represented by its kmer content

Atomic question

- Given a queried kmer, <u>does it exist</u> in the indexed set?



Genomic research engine: conceptual view

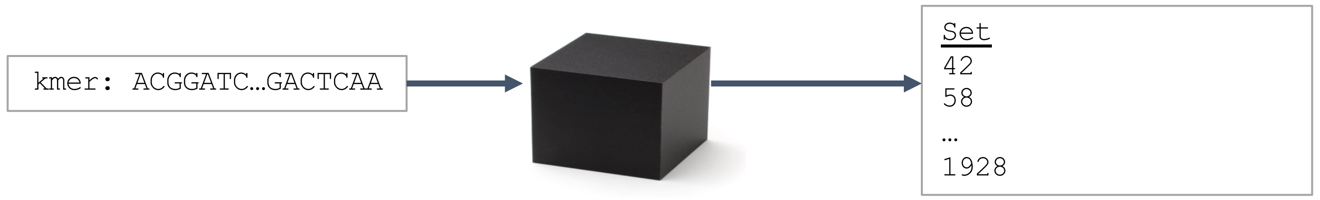
index

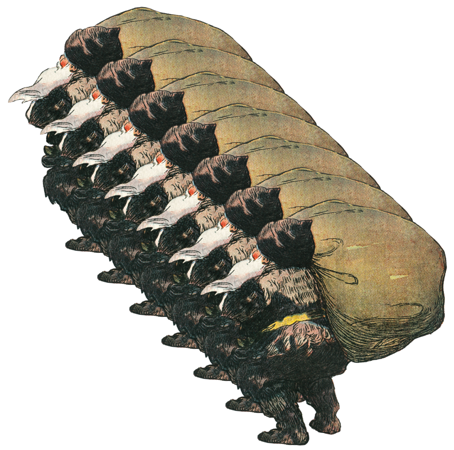
Set representation

- A bank (genome, reads, …) represented by its kmer content

Atomic question

- Given a queried kmer, in <u>which sets</u> does it exist?





Genomic research engine: conceptual view

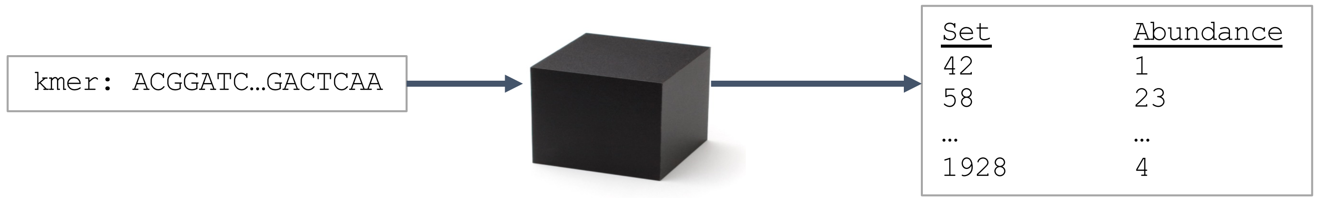
index

Set representation

- A bank (genome, reads, …) represented by its kmer content

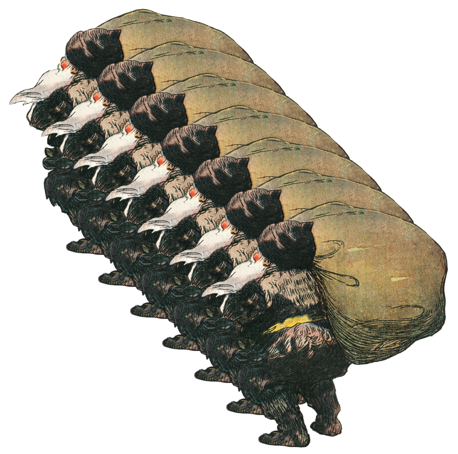
Atomic question

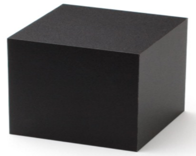
- Given a queried kmer, in which sets, with which abundance?



```
kmer: ACGGATC…GACTCAA
```

Set	Abundance
42	1
58	23
…	…
1928	4





: A bloom filter

Bloom Filter

A bit vector B of fixed size

1	0	0	1	0	0	1	1

Add one element -> hash(element) -> set B[hash] = 1

Query one element -> hash(element) -> returns B[hash]

0: absent

1: present (**possibly a False Positive**)

counting Bloom Filter

A bit vector B of fixed size, x bits per element

Add one element -> hash(element) -> B[hash] += 1

Query one element -> hash(element) -> returns B[hash]

0: absent

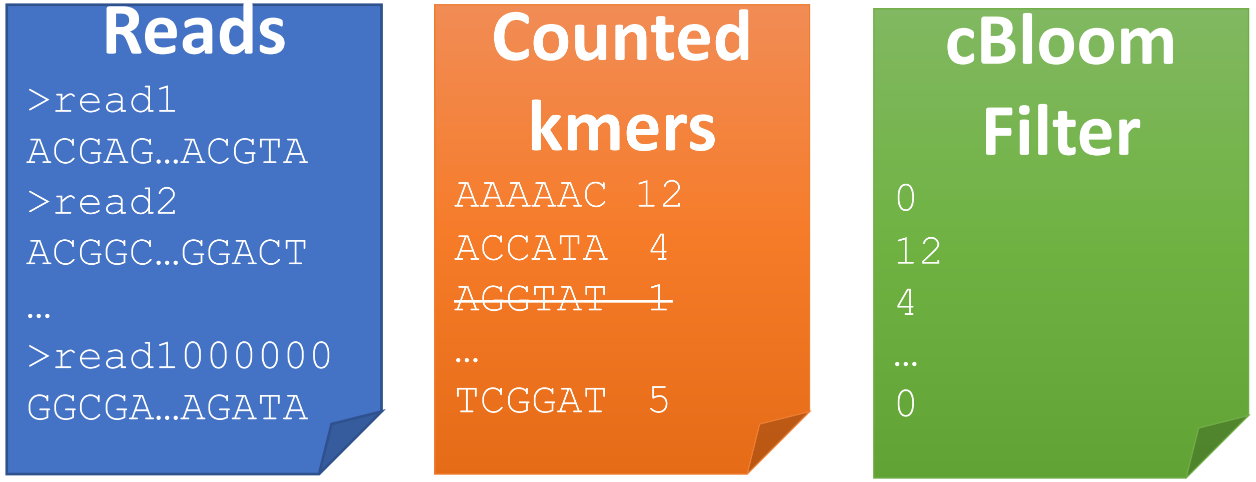
n>0: present with non null False Positive Rate

n: abundance (**possibly overestimated**)

Indexing: conceptual view

One read set:

- Extract & count **kmers**
- Filter kmers
- Generate a [counting] bloom filter



Reads

```
>read1
ACGAG…ACGTA
>read2
ACGGC…GGACT
…
>read1000000
GGCGA…AGATA
```

Counted kmers

```
AAAAAC 12
ACCATA  4
~~AGGTAT  1~~
…
TCGGAT  5
```

cBloom Filter

```
0
12
4
…
0
```

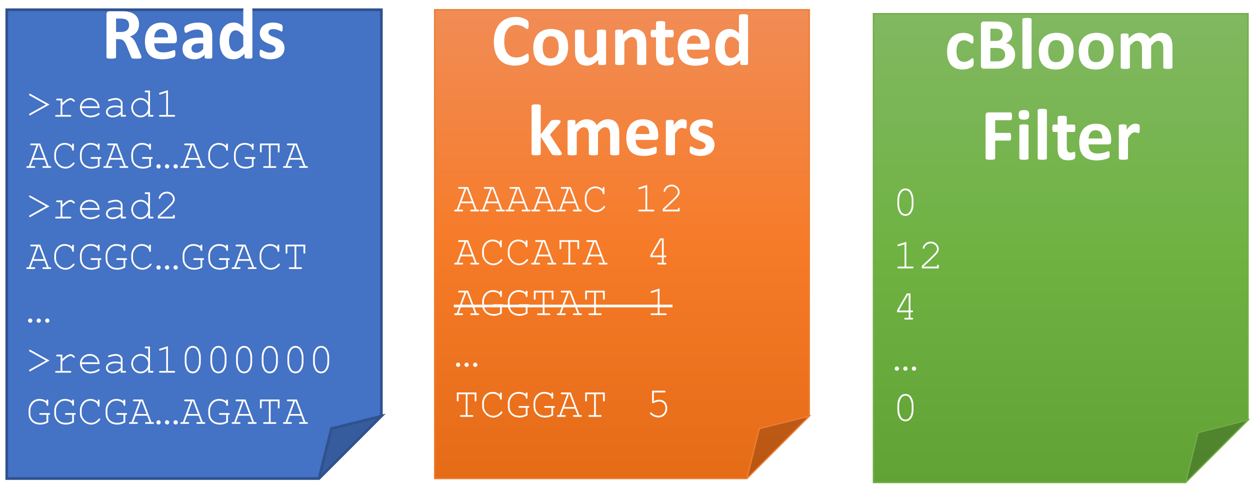
Indexing: conceptual view

One read set:

- Extract & count **kmers**
- Filter kmers
- Generate a [counting] bloom filter

N read sets:

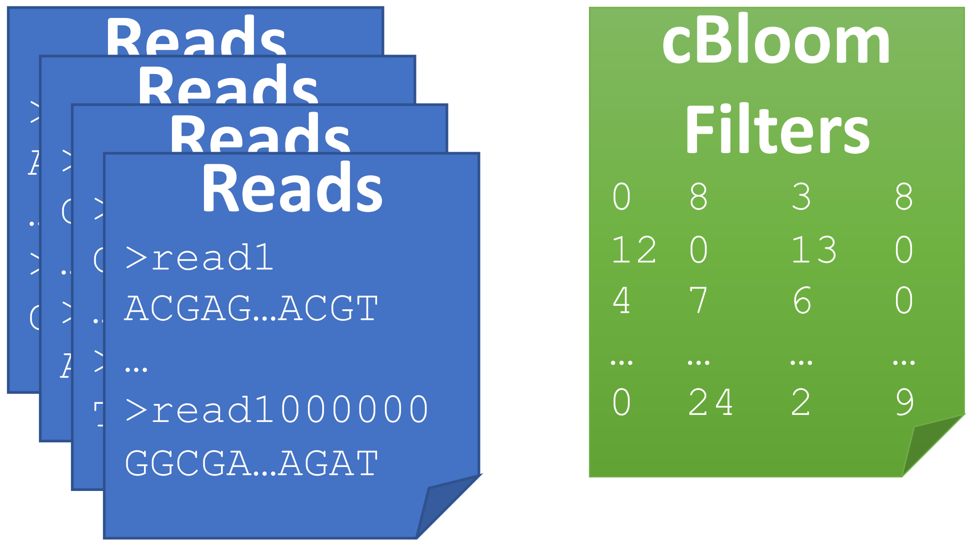
- Create N [counting] bloom filters
- This is the index



```
>read1
ACGAG…ACGTA
>read2
ACGGC…GGACT
…
>read1000000
GGCGA…AGATA
```

```
AAAAAC  12
ACCATA   4
```
~~AGGTAT 1~~
```
…
TCGGAT   5
```

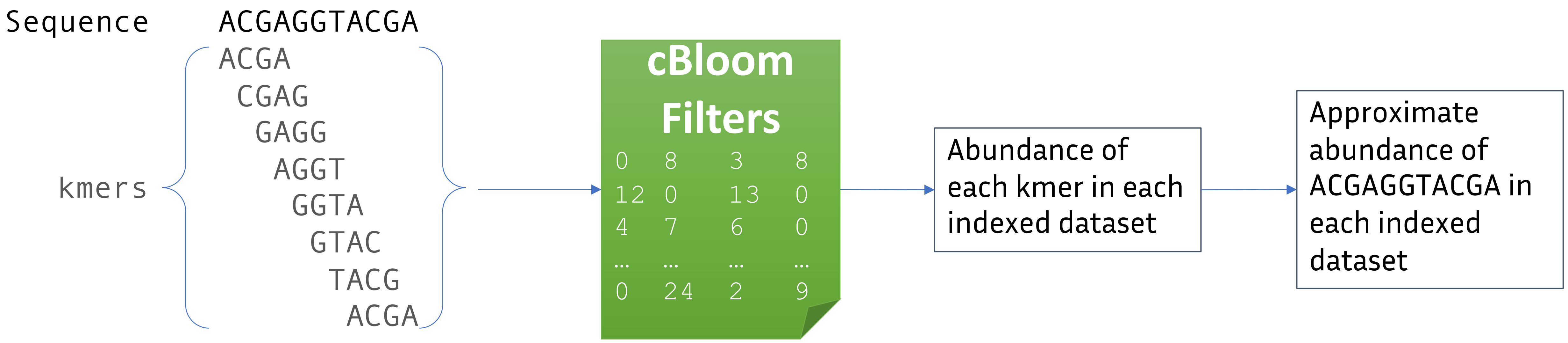
```
0
12
4
…
0
```



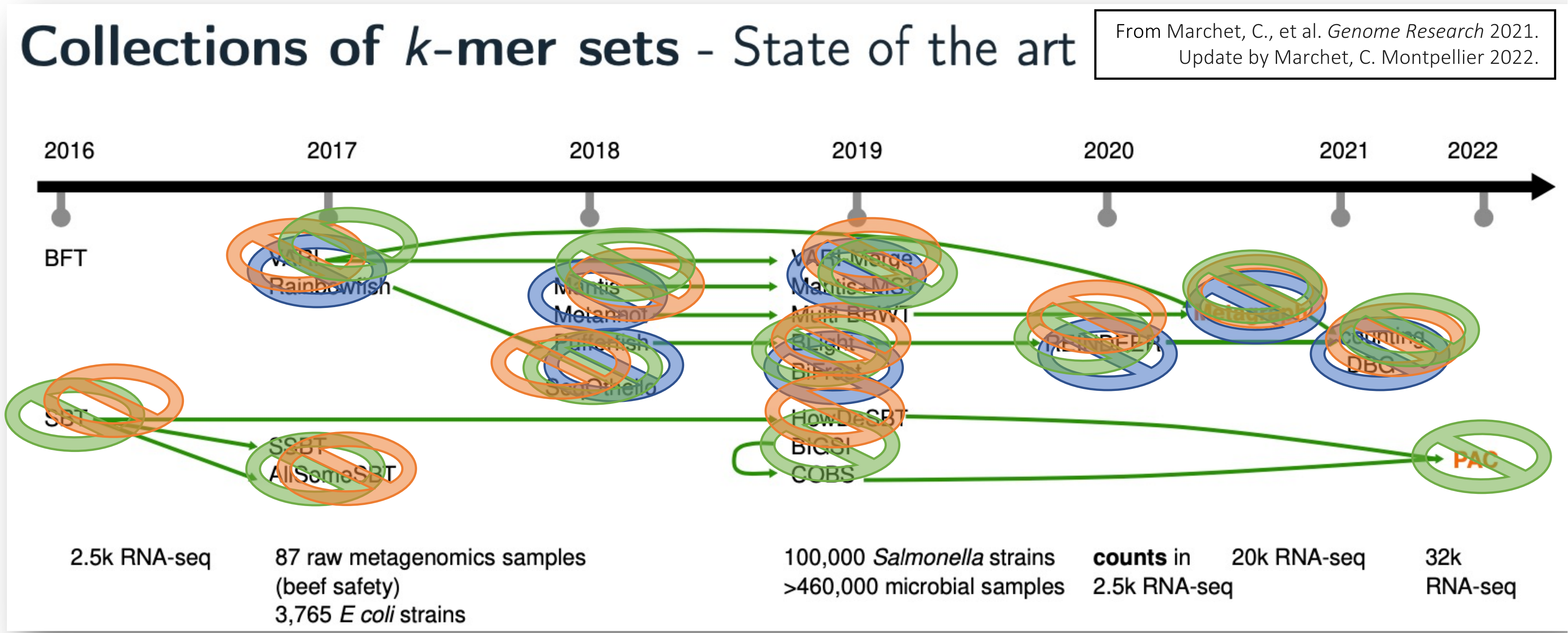
```
>read1
ACGAG…ACGT
…
>read1000000
GGCGA…AGAT
```

0	8	3	8
12	0	13	0
4	7	6	0
…	…	…	…
0	24	2	9

Querying: conceptual view



Possible today ?



Fast (indexing + query)
- No random access

→ no tree

Note
theoretical complexity != practical speed (modern compilers)

-03

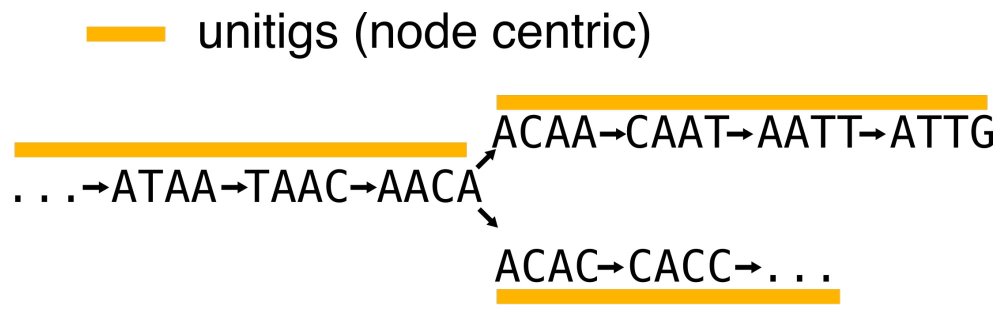
Low memory
- Do not raise index in RAM at indexing or query time

High complexity
- Do not require unitigs (next slides)
- No not associate k-mers to color vector

Note (even more knowing that many of the authors of these tools are in the room ☺)
These tools are not adapted to our objectives (prev. slide) but
Adapted to other contexts, and have other features (alignments, variants, …)

Other tools:
- **ggcat** – endless query (debug in progress)
- **Needle** – Not precise enough
- **PebbleScout** – endless indexing

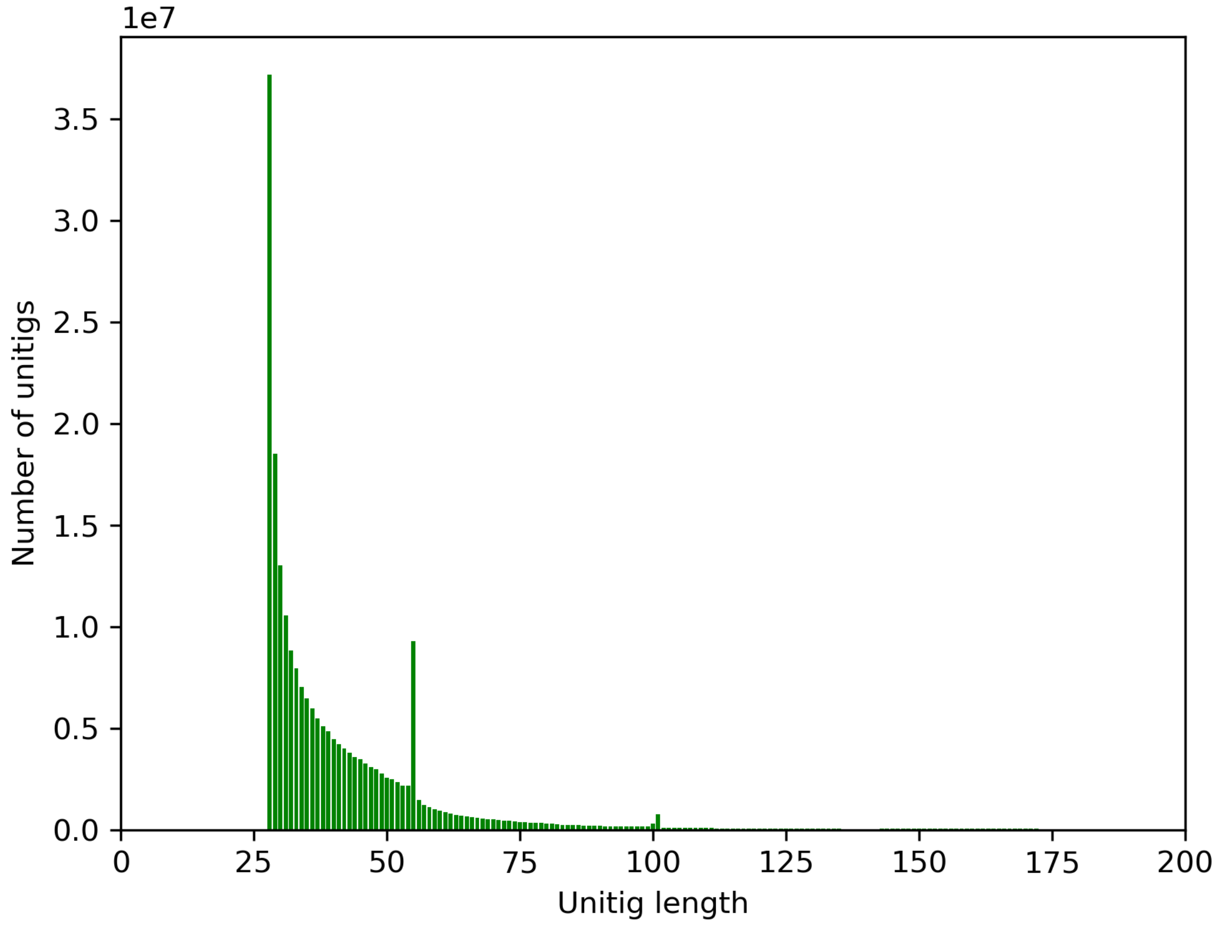
Figure from Marchet, C., Sneak peek at the -tigs!



Cannot use unitigs

- On complex data, unitigs do not help

- Example on a Tara Ocean sample (complex marine metagenome), k=28
  - 213 millions unitigs
  - average length 42
  - Requires 3h20 to compute (bcalm [1])



[1] Chikhi, R., Limasset, A., & Medvedev, P. (2016). Compacting de Bruijn graphs from sequencing data quickly and in low memory. *Bioinformatics*, *32*(12), i201-i208.

Proposed solution

fimpera + kmindex

Two contributions

[Counting] Bloom Filters

Lucas Robidou

- Exponential decrease of Bloom filter FPrate
- Decrease of counting Bloom Filter overestimations

Robidou, L, Peterlongo P. "fimpera: drastic improvement of Approximate Membership Query data-structures with counts." Bioinformatics 39.5 (2023) doi: https://doi.org/10.1101/2022.06.27.497694

From reads to Indexes

Téo Lemane

- Optimized kmer index:
  - Representation
  - Creation
  - Update
  - Query

Lemane T. et. al. *“kmindex and ORA: indexing and real-time user-friendly queries in terabytes-sized highly complex genomic datasets”* https://www.biorxiv.org/content/10.1101/2023.05.31.543043v2

Lucas Robidou https://github.com/lrobidou/fimpera

z = k-s

Fimpera: Counting BF with low disk, low FP, low counting overestimates, no drawback

key idea for presence absence:

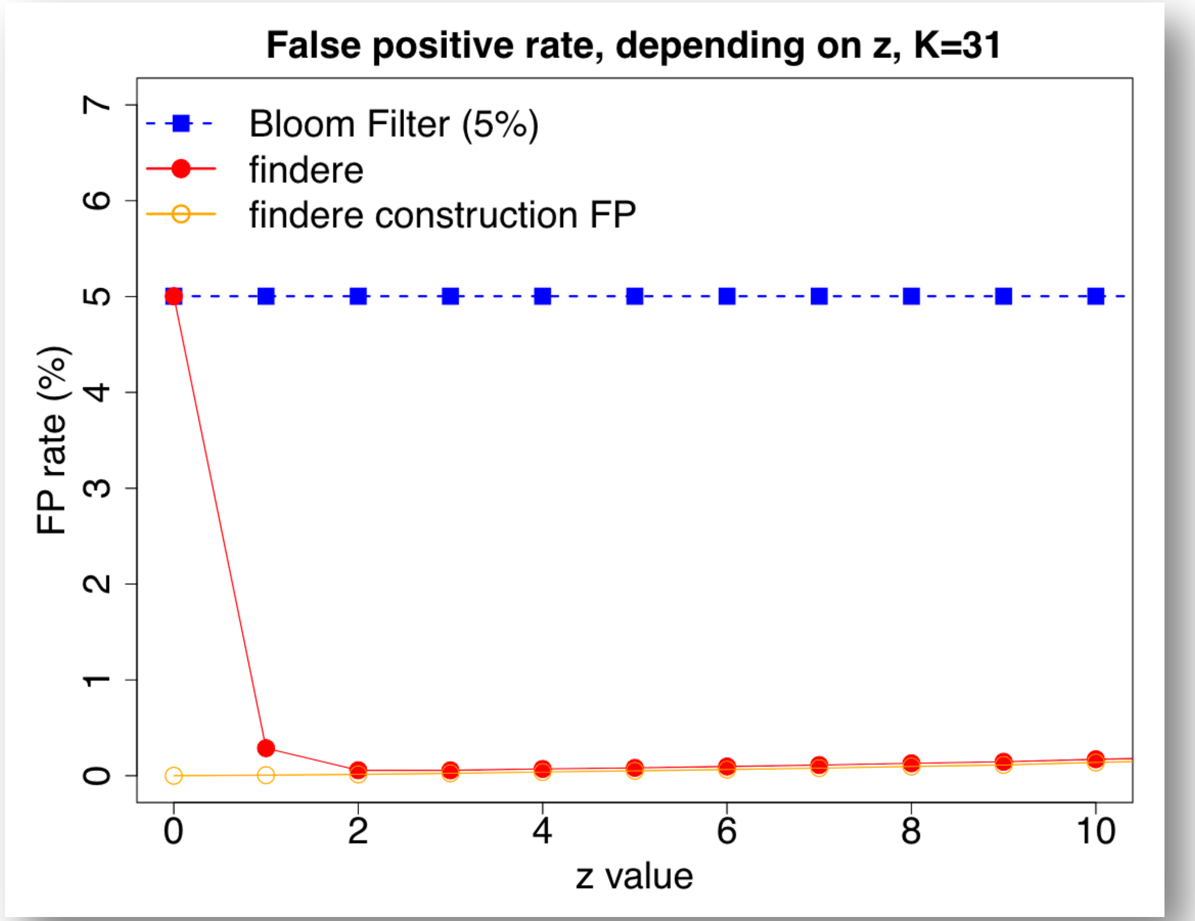
- If a kmer exists all words inside this kmer (smers) exist

$\longleftrightarrow$

- If a smer of a kmer does not exist, the kmer does not exist

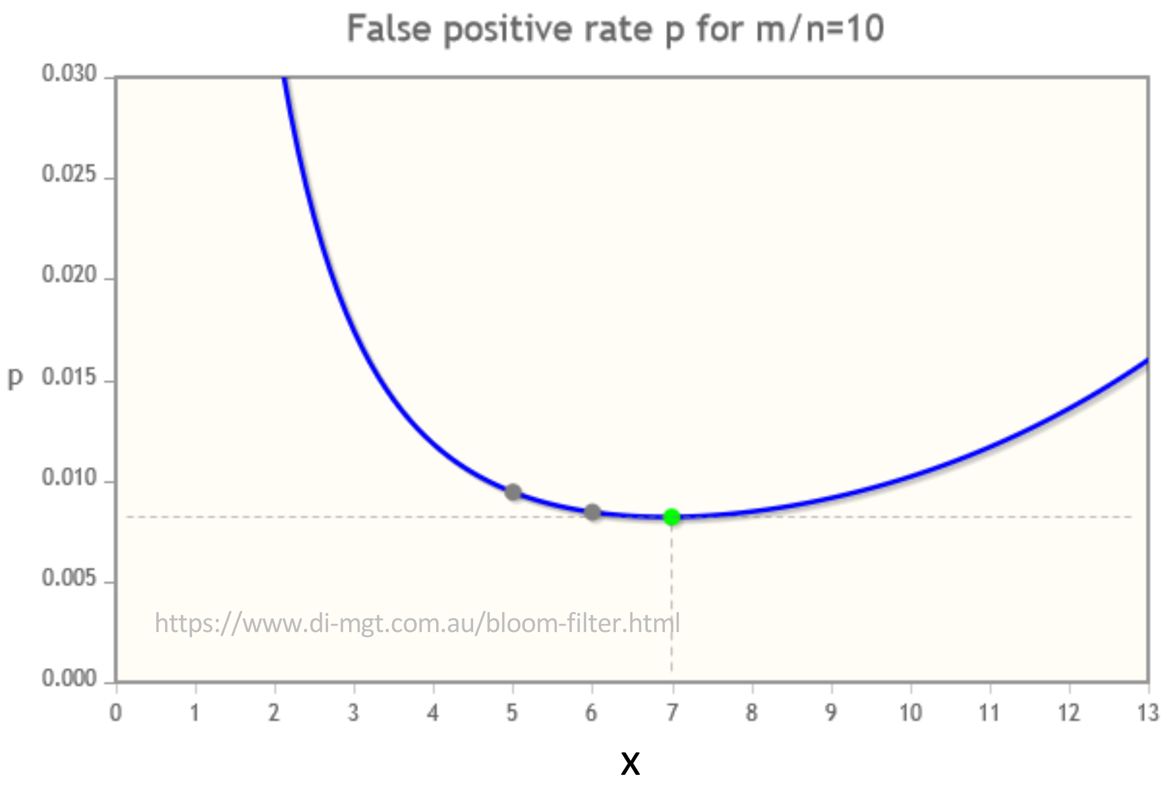
In practice:

- Index smers
- When querying a kmer, report it as present *iif* all its constituent smers are present



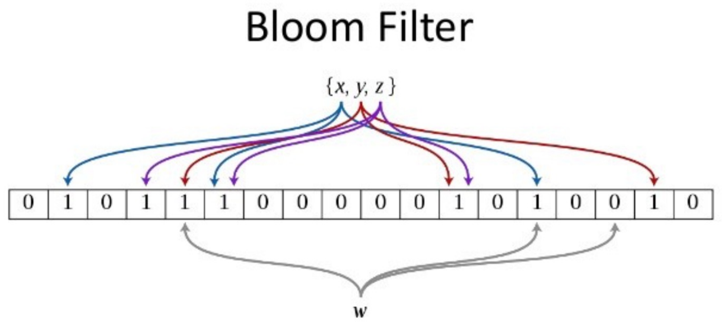
Lucas Robidou https://github.com/lrobidou/fimpera

Fimpera: compared to x hash functions in a Bloom Filter

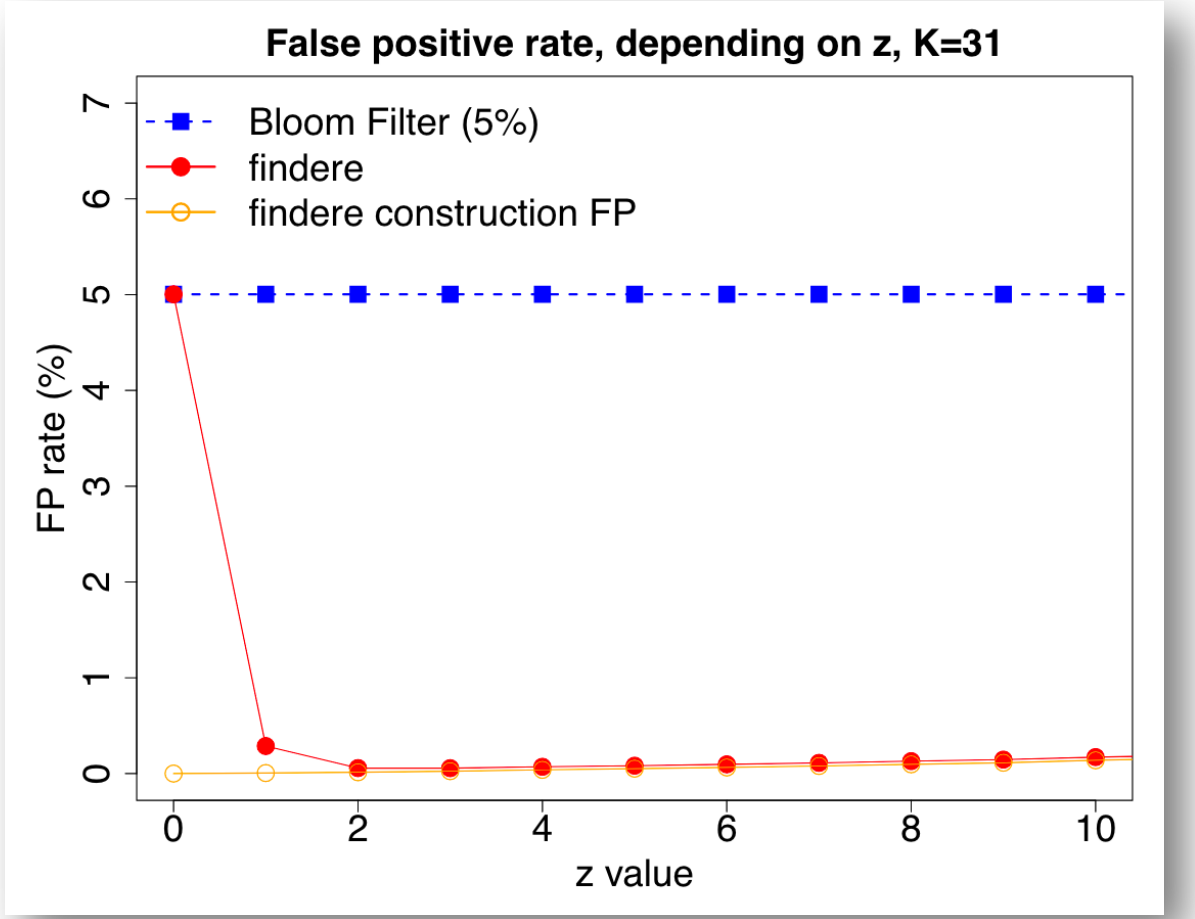


- ☹ x accesses
- ☹ Caches misses
- ☹ Saturates the filter

David Eppstein — self-made, originally for a talk at WADS 2007



z = k-s



z = k-s

Fimpera: Counting BF with low disk, low FP, low counting overestimates, no drawback

key idea for presence absence:

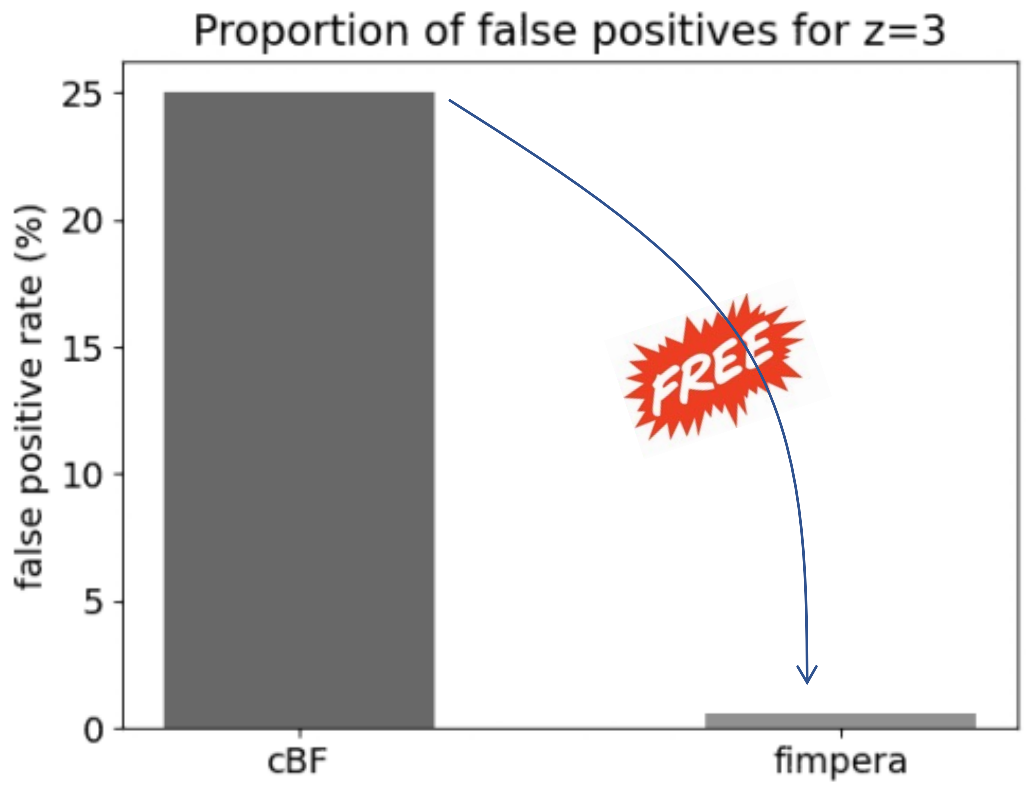
- If a kmer exists all words inside this kmer (smers) exist

$\longleftrightarrow$

- If a smer of a kmer does not exist, the kmer does not exist

In practice:

- Index smers
- When querying a kmer, report it as present *iif* all its constituent smers are present



Indexed: Tara Ocean ERR1726642
Queried: Tara Ocean ERR4691696

Fimpera: Counting BF with low disk, low FP, low counting overestimates, no drawback

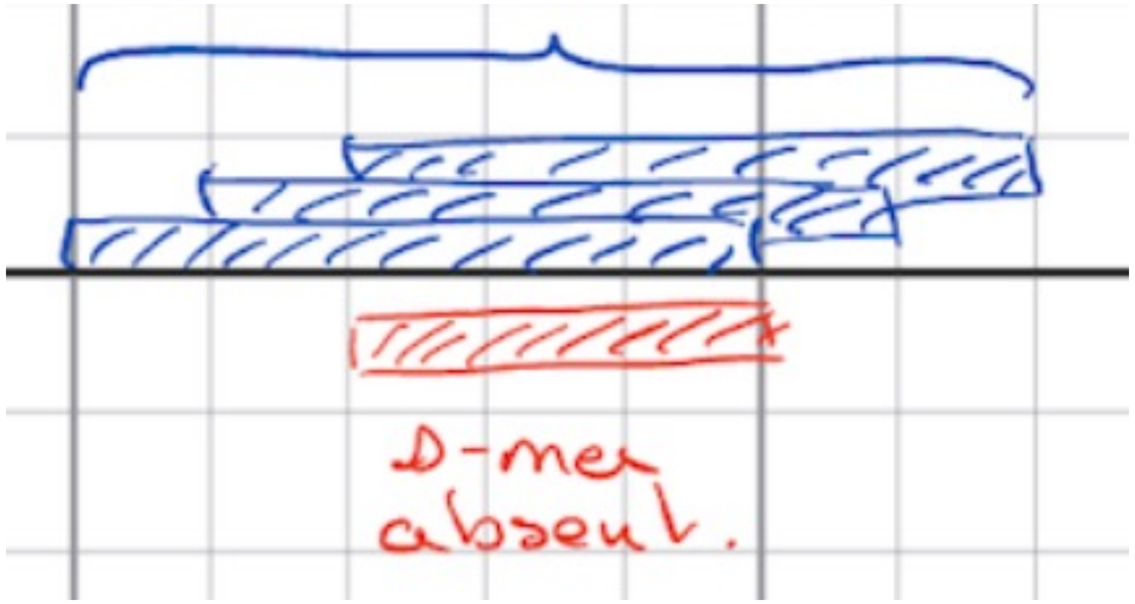
Save query time

key idea for presence absence:

- If a kmer exists all words inside this kmer (smers) exist

⟷

- If a smer of a kmer does not exist, the kmer does not exist



In practice:

- Index smers
- When querying a kmer, report it as present *iif* all its constituent smers are present

Lucas Robidou https://github.com/lrobidou/fimpera

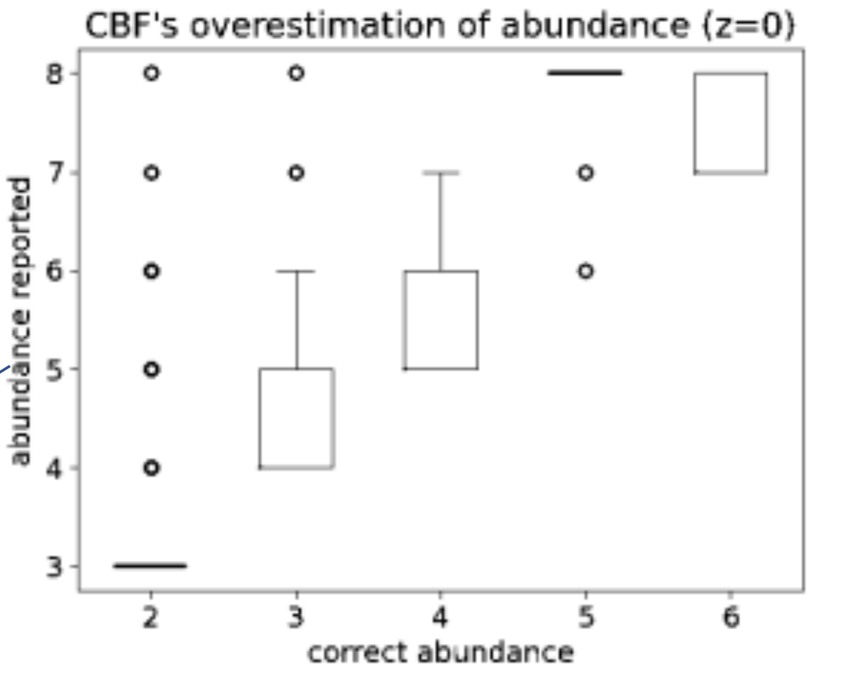
Fimpera: Counting BF with low disk, low FP, low counting overestimates, no drawback

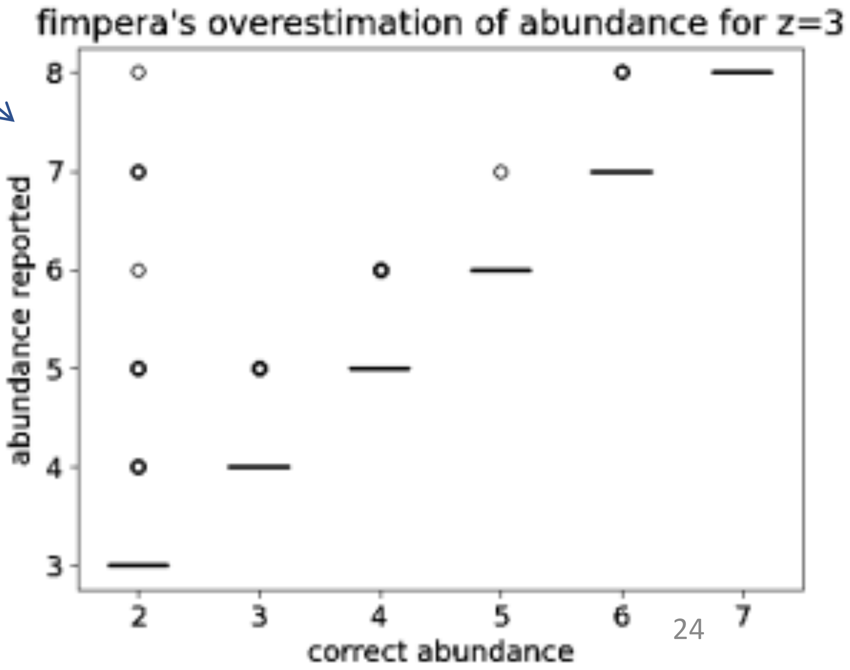
key idea for abundance:

- The abundance of a kmer is at most the abundance of its less abundant constituent smer

In practice:

- Index smers abundances
- When querying a kmer, return the abundance of its least abundant smer





Indexed: Tara Ocean ERR1726642
Queried: Tara Ocean ERR4691696

Lucas Robidou https://github.com/lrobidou/fimpera

Fimpera: Counting BF with low disk, low FP, low counting overestimates, no drawback

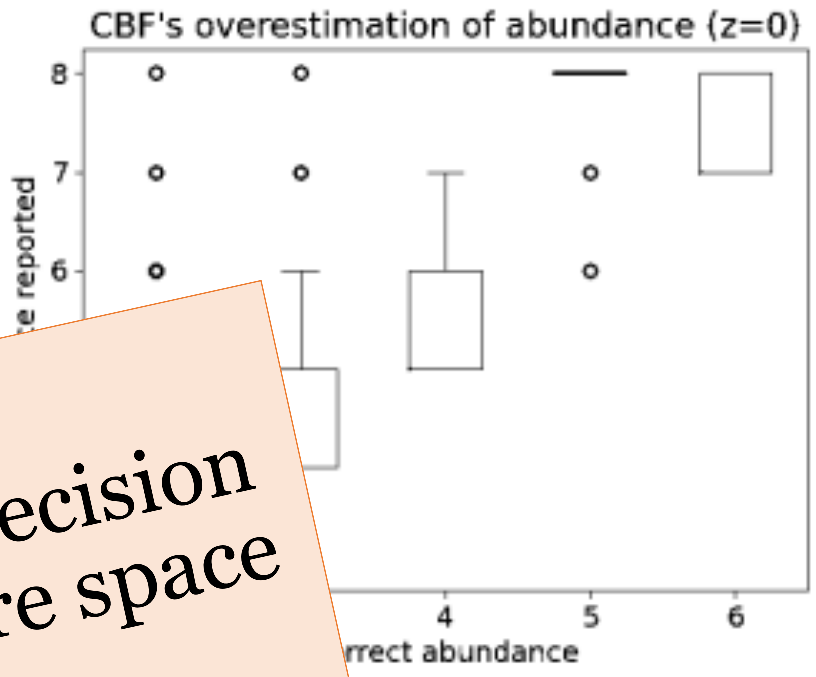
key idea for abundance:

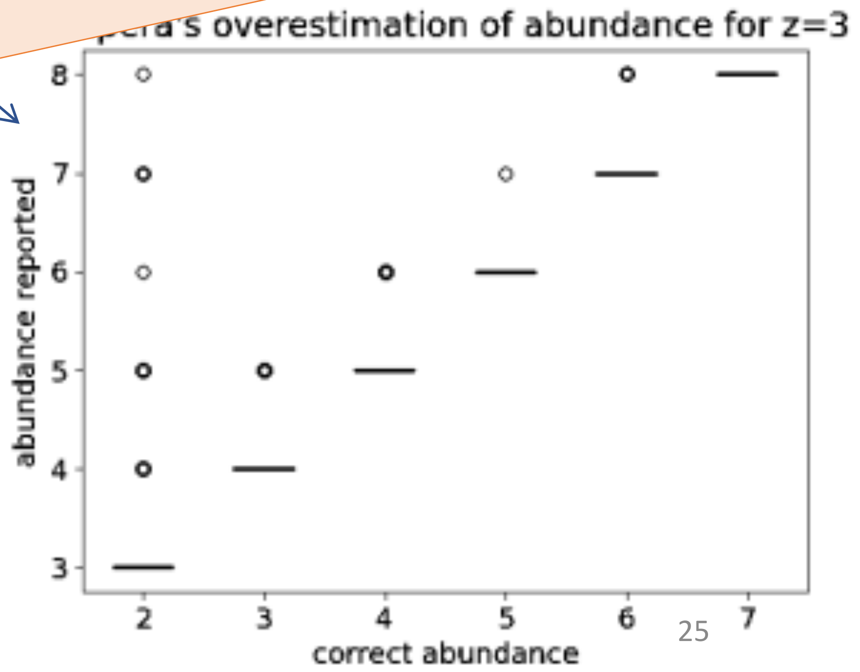
- The abundance of a kmer is a
  abundance of its l
  constitu

In practice

- Index sme
- When que ing a kmer, return the abundance of its least abundant smer

Without Fimpera, the same precision would require ~35x times more space



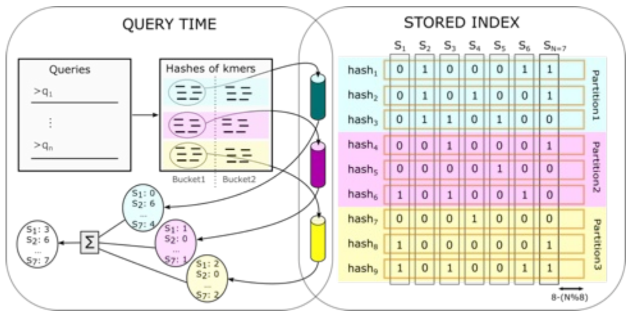


Indexed: Tara Ocean ERR1726642
Queried: Tara Ocean ERR4691696

Téo Lemane

https://github.com/tlemane/kmindex

Kmindex: indexation and real-time query of kmers in terabyte-sized genomic data banks



key features

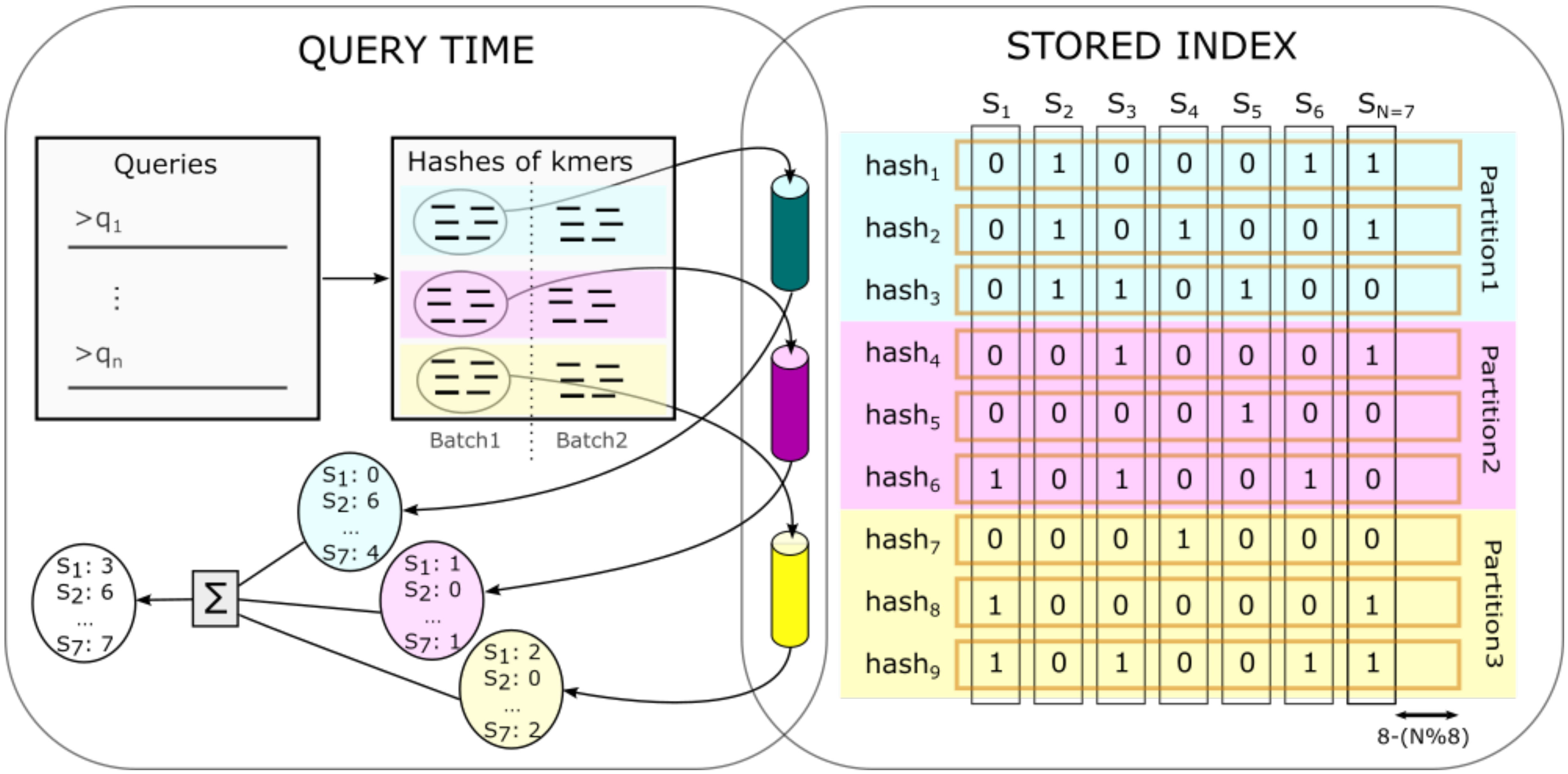
- **1st of all:**
  - **kmers -> sorted hash values**
- Count hashes instead of ascii kmers
- Clever kmer filtration process
- Kmers -> partition -> parallelization
  - At indexing time
  - At query time

key features

- simple inverted index
- mmap at query time
  - Clever pages in RAM management
- Integrates Fimpera

Based on kmtricks :

Lemane, T., Medvedev, P., Chikhi, R., & Peterlongo, P. Bioinformatics Advances, 2(1), vbac029.



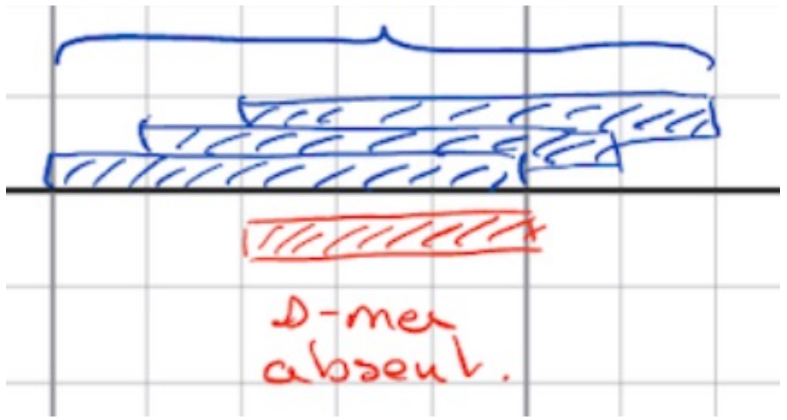
	S_1	S_2	S_3	S_4	S_5	S_6	$S_{N=7}$
$hash_1$	0	1	0	0	0	1	1
$hash_2$	0	1	0	1	0	0	1
$hash_3$	0	1	1	0	1	0	0
$hash_4$	0	0	1	0	0	0	1
$hash_5$	0	0	0	0	1	0	0
$hash_6$	1	0	1	0	0	1	0
$hash_7$	0	0	0	1	0	0	0
$hash_8$	1	0	0	0	0	0	1
$hash_9$	1	0	1	0	0	1	1

kmindex main technical ideas

Avoid branching in the code

- Fimpera: can skip query positions
- Theoretical advantage

```
if (v[i] == false)
      i = i+s;
```

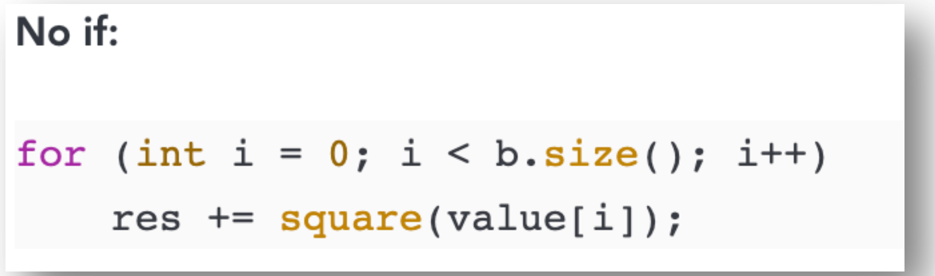


kmindex main technical ideas

Avoid branching in the code

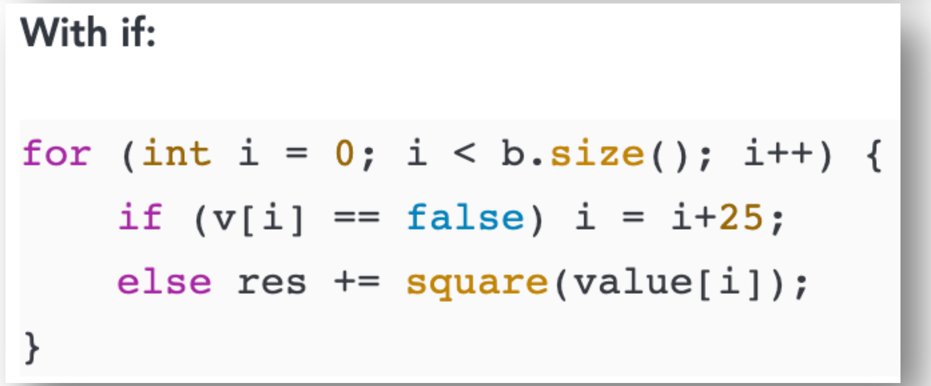
- Fimpera: can skip query positions
- Good idea? – small experiment

b:
- 1 billion Booleans
- 10% are ”false” randomly distributed



No if:

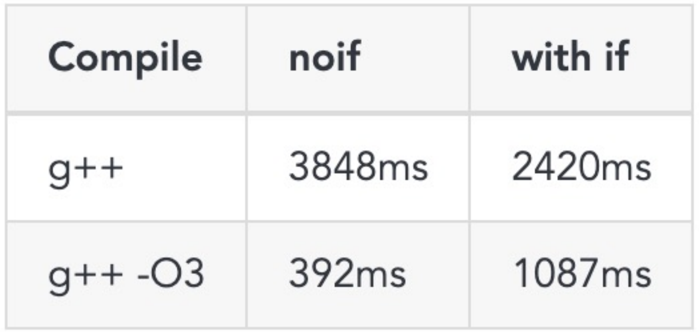
```
for (int i = 0; i < b.size(); i++)
    res += square(value[i]);
```



With if:

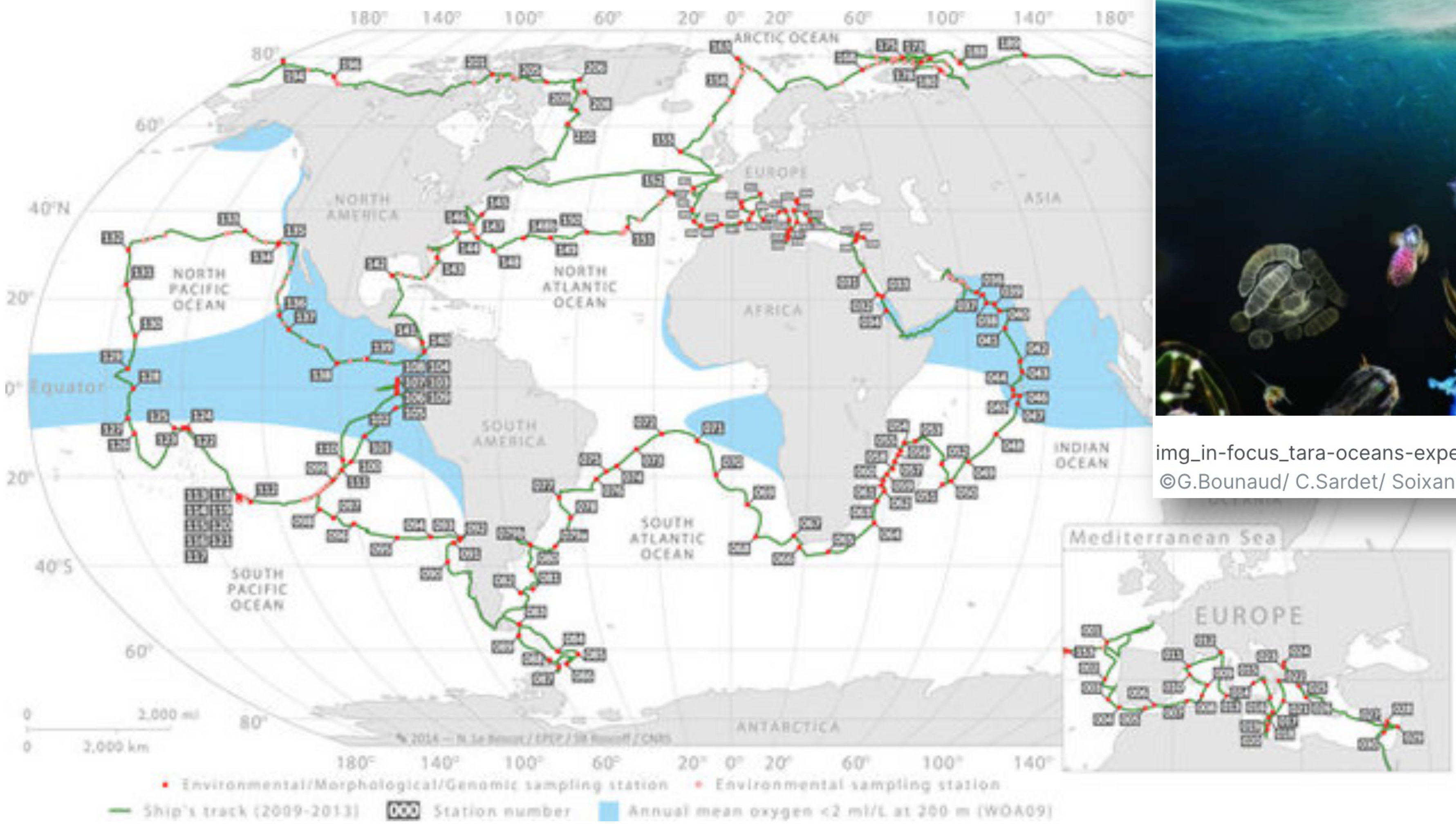
```
for (int i = 0; i < b.size(); i++) {
    if (v[i] == false) i = i+25;
    else res += square(value[i]);
}
```

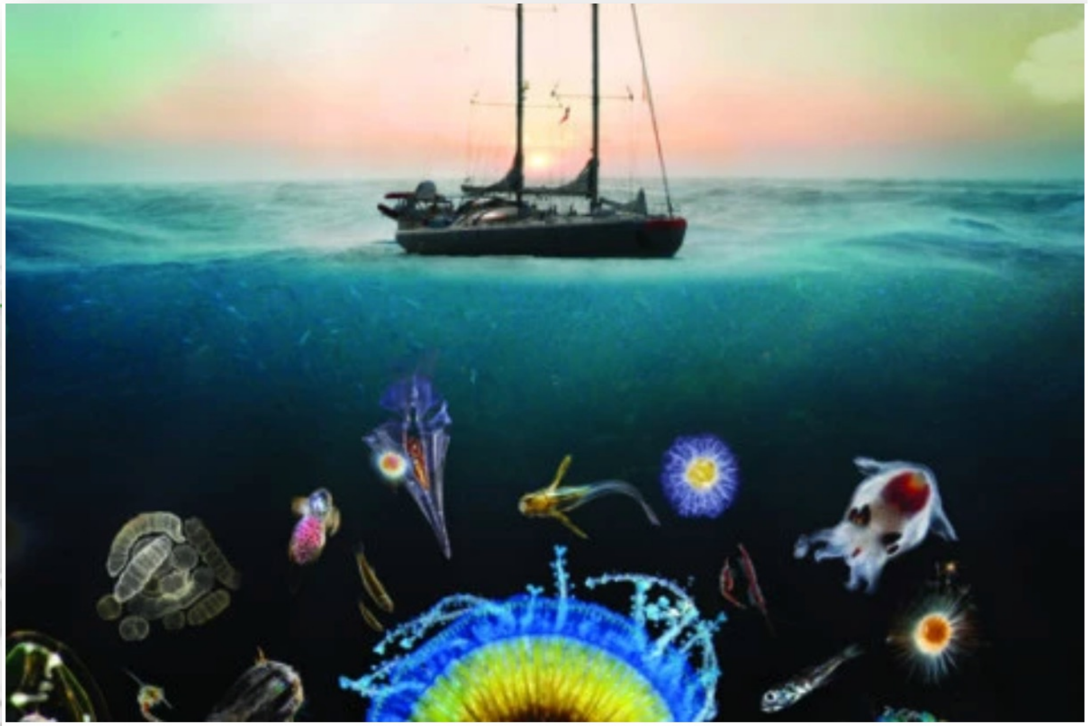
~5.2 times less calls to “square”



Compile	noif	with if
g++	3848ms	2420ms
g++ -O3	392ms	1087ms

RESULTS





img_in-focus_tara-oceans-expeditions-plankton
©G.Bounaud/ C.Sardet/ Soixanteseize/ Tara Expéditions

Pesant, S., Not, F., Picheral, M., Kandels-Lewis, S., Le Bescot, N., Gorsky, G., ... & Searson, S. (2015). Open science resources for the discovery and analysis of Tara Oceans data. *Scientific data*, *2*(1), 1-16.

Result: Index construction

Databank:

- 50 Tara Ocean samples
- Avg 11 billions distinct kmers per sample
- 1.4TB fastq.gz

Indexing: one command line

```
kmindex files      |smer|            |bloom|
                    (23)          (30billions)
```

- Wall clock time: **2h56**
- Peak RAM: 107GB
- Peak disk: 878GB
- Final index size: 164GB

Tara Schooner - Creative Commons Attribution 3.0

Result: query

Databank:

- 50 Tara Ocean samples
- Avg 11 billions distinct kmers per sample
- 1.4TB fastq.gz

querying: one command line: `` `kmindex query index query.fa` ``

#queries (reads)	1	10k	1 million	10 millions
Max RAM (GB)	0.005	0.05	4.9	46.7
Time (s) – cold RAM	<0.1	20	94	261 (4m21s)
Time (s) – warm RAM	<0.1	10.84	41	227

“rocket mode”
Use as much RAM as available

#queries (reads)	1	10k	1 million	10 millions
Max RAM (GB)	0.005	2.84	133	194
Time (s) – cold RAM	<0.1	17	61	99
Time (s) – warm RAM	<0.05	7	16	64

Comparative results

Databank:

- 50 Tara Ocean samples
- Avg 11 billions distinct kmers per sample
- 1.4TB fastq.gz

Build

	Step	Wall clock time	Max Memory (GB)	Max temp. disk (GB)	Output size on disk (GB)
MetaGraph	**KMC3** count	3h44	278	1019	1019
	KMC3 dump	18h11	0	5684	5684
	Build	71h30	250	1580	531
	Overall	93h40	278	5684	531
MetaProFi	**KMC3** count	3h44	278	1019	1019
	KMC3 dump	18h11	0	5684	5684
	MetaProFi	8h20	232	226	226
	Overall	30h15	278	5684	226
PAC	All	15h59	190	$191 + 1415^{\beta}$	184
kmindex	All	**2h56**	**107**	**878**	**164**

$^{\beta}$ in order to consider multiple files per sample, the original input file has to be concatenated and so doubled using **PAC**.

Query

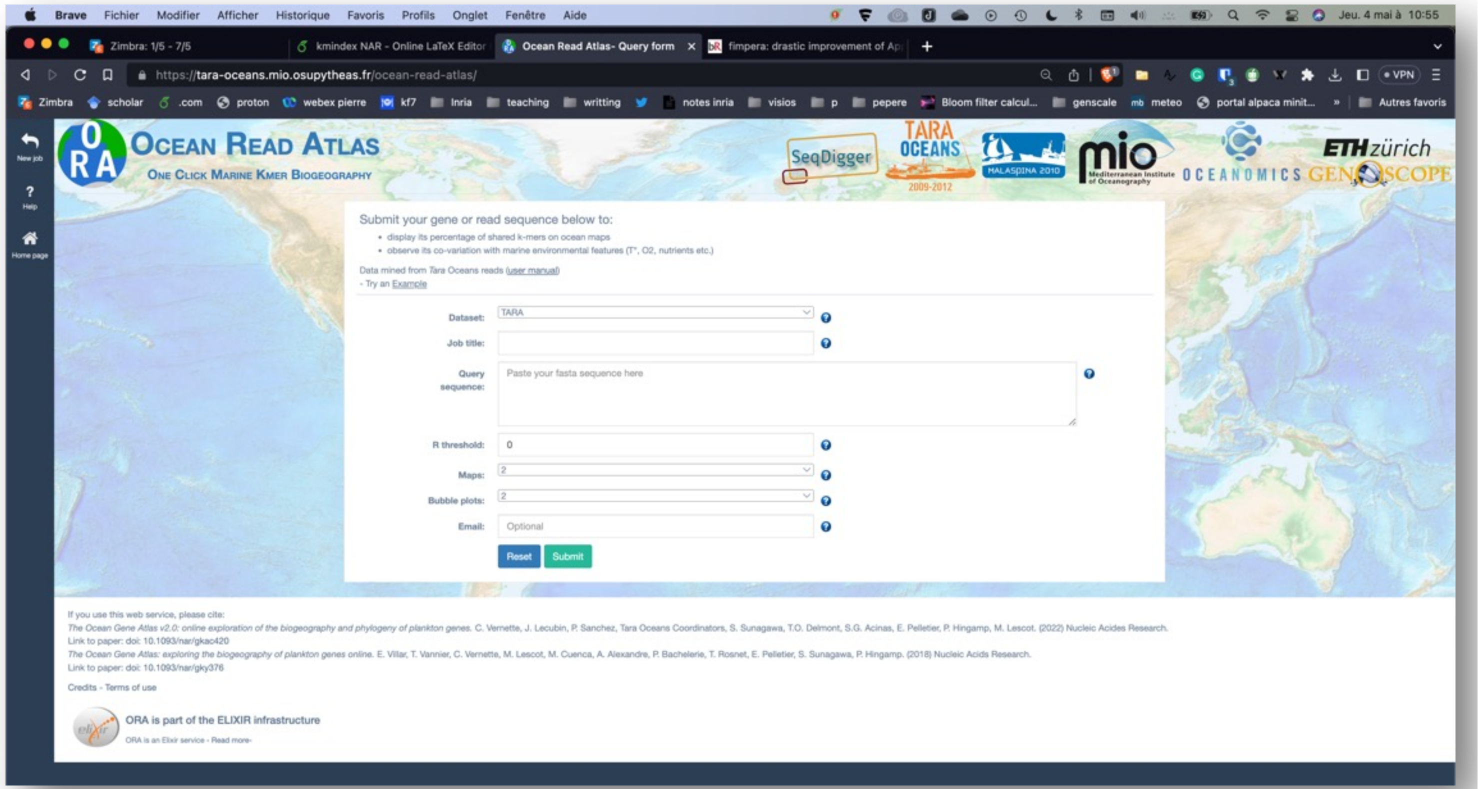
No. queries	1	10	100	1,000	10,000	100,000	1,000,000	10,000,000
MetaGraph Time	58m46							
MetaGraph Memory peak (GB)	148							
MetaProFi Time	0m12	0m15	1m33	2m57	3m02	3m37	11m56	1h29m12
MetaProFi Memory peak (GB)	0.3	0.3	0.3	0.32	0.44	2.25	21	203
PAC Time	5m30	16m48	34m31	38m58	36m06	36m03	39m54	36m35
PAC RAM (GB)	89	90	90	90	90	90	92	104
kmindex Time	**0s06**	**0s23**	**1s24**	**4s71**	**19s78**	**53s72**	**1m13s**	**4m21s**
kmindex Memory peak (GB)	**0.005**	**0.005**	**0.006**	**0.01**	**0.05**	**0.45**	**4.9**	**46.7**

ORA Server

https://ocean-read-atlas.mio.osupytheas.fr/

Index: all Tara Ocean Metagenomic samples (no abundance yet)

- Input fastq.gz files
  - 282 TB
  - 1,393 samples
- Final index size: 36TB
- Each sample:
  - Position
  - Species fraction sizes
  - Physico-chemical env.:
    - Ph, salinity, T°, …



Brave Fichier Modifier Afficher Historique Favoris Profils Onglet Fenêtre Aide
Jeu. 4 mai à 10:55
Zimbra: 1/5 - 7/5
kmindex NAR - Online LaTeX Editor
Ocean Read Atlas- Query form
fimpera: drastic improvement of Ap
https://tara-oceans.mio.osupytheas.fr/ocean-read-atlas/
Zimbra scholar .com proton webex pierre kf7 Inria teaching writting notes inria visios p pepere Bloom filter calcul... genscale meteo portal alpaca minit... Autres favoris
New job
Help
Home page
OCEAN READ ATLAS
ONE CLICK MARINE KMER BIOGEOGRAPHY
SeqDigger
TARA OCEANS 2009-2012
MALASPINA 2010
mio Mediterranean Institute of Oceanography
OCEANOMICS
GENOSCOPE
ETH zürich

Submit your gene or read sequence below to:

- display its percentage of shared k-mers on ocean maps
- observe its co-variation with marine environmental features (T°, O2, nutrients etc.)

Data mined from *Tara* Oceans reads (user manual)
- Try an Example

Dataset:	TARA
Job title:	
Query sequence:	Paste your fasta sequence here
R threshold:	0
Maps:	2
Bubble plots:	2
Email:	Optional

Reset Submit

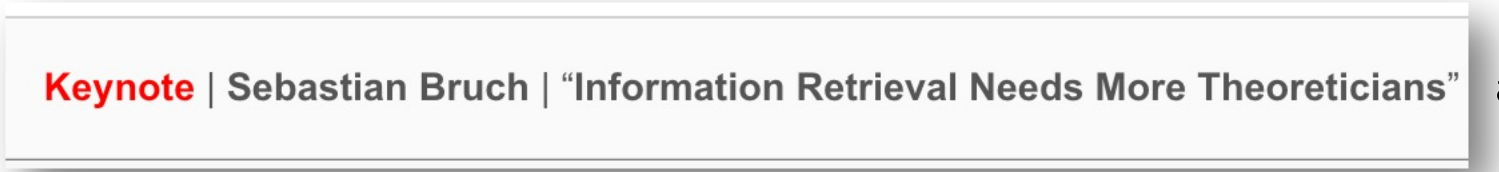
If you use this web service, please cite:
The Ocean Gene Atlas v2.0: online exploration of the biogeography and phylogeny of plankton genes. C. Vernette, J. Lecubin, P. Sanchez, Tara Oceans Coordinators, S. Sunagawa, T.O. Delmont, S.G. Acinas, E. Pelletier, P. Hingamp, M. Lescot. (2022) Nucleic Acides Research.
Link to paper: doi: 10.1093/nar/gkac420
The Ocean Gene Atlas: exploring the biogeography of plankton genes online. E. Villar, T. Vannier, C. Vernette, M. Lescot, M. Cuenca, A. Alexandre, P. Bachelerie, T. Rosnet, E. Pelletier, S. Sunagawa, P. Hingamp. (2018) Nucleic Acids Research.
Link to paper: doi: 10.1093/nar/gky376

Credits - Terms of use

ORA is part of the ELIXIR infrastructure
ORA is an Elixir service - Read more-

To conclude

Take home messages


Keynote | Sebastian Bruch | "Information Retrieval Needs More Theoreticians"

and practical considerations

If you want to scale up to big and complex data, consider:

- Limited access to RAM
- No cache misses
- Avoid branching in code

Suggestion:
Include these considerations in addition to theoretical complexity analyses

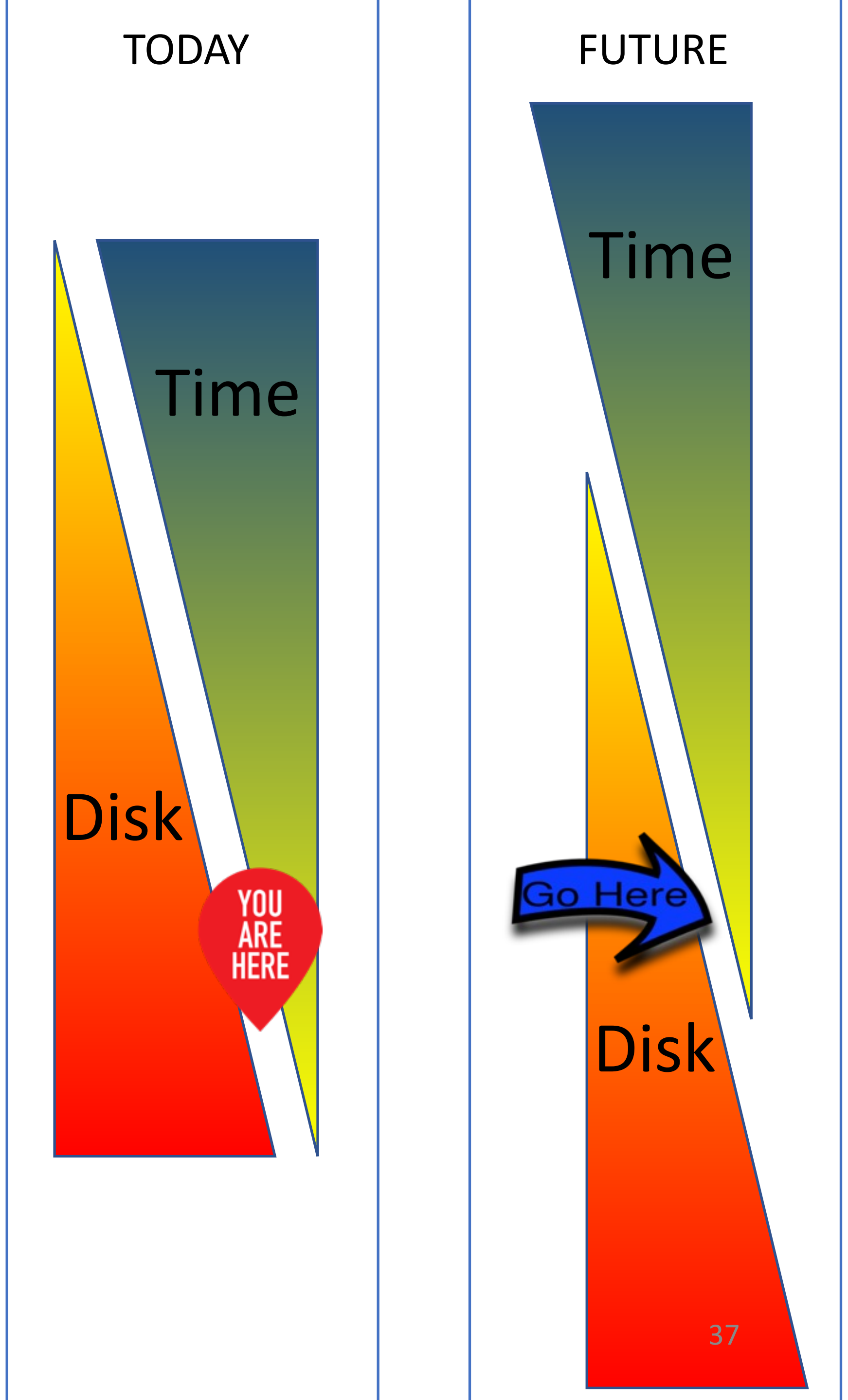
"compiled" complexity

Limitations

- Not adapted to many “simple” samples
- Requires fast and local disk

What comes next?

- **From TB to PB**
  - **Less disk <-> More Time**
- Answers from 0.01s to 0.1s is ok
- Compress lines of the inverted indexes
  - RRR, LZ, grammars, … ?



https://github.com/tlemane/kmindex
https://github.com/lrobidou/fimpera

https://ocean-read-atlas.mio.osupytheas.fr/

Thanks ☺