Compressed indexation structure for analysing collections of similar genomes

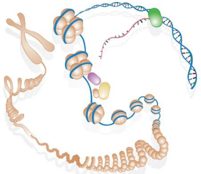
Annie CHATEAU & Alban MANCHERON (LIRMM, équipe MAB),
Gautier SARAH, Gaetan DROC & Manuel RUIZ (CIRAD, UMR AGAP, équipe ID/CIAT)

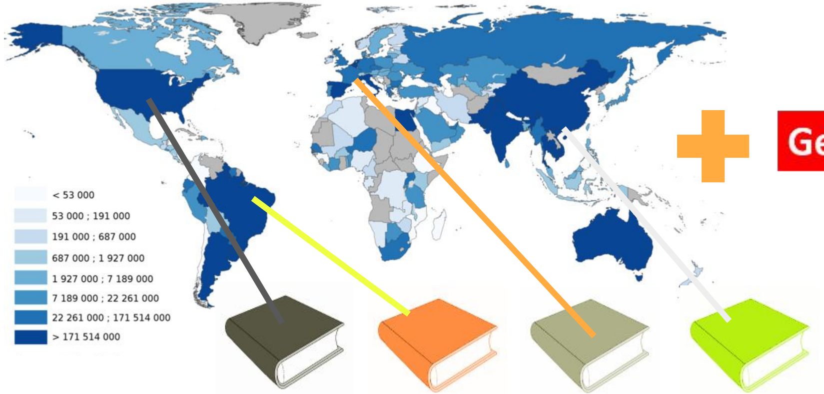
Clément AGRET

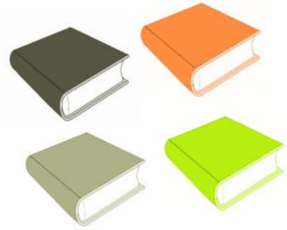
<u>Équipes</u> : ID : Intégration Données
MAB : Méthodes et algorithmes pour la bioinformatique

Data & Computation

Let's simplify, what’s happen if we see the GENOME as a simple book ?

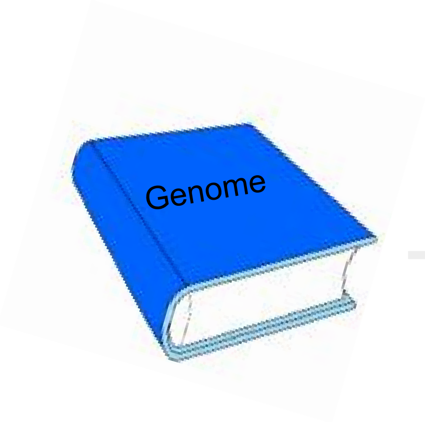
If  = 



\+ GenomeHarvest = 

A lot of books to read !!

Data & Computation



SuperBook

1
BOOK

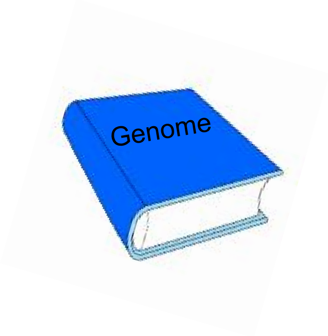
+1000
BOOKS

If all books tell more or less the same story, what about writing a SuperBook ?

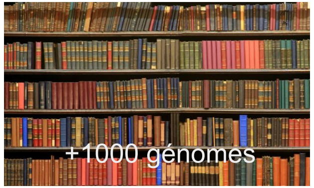
Data & Computation

When you read a book you should be able to answer to questions like:

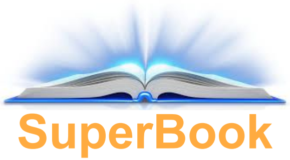
- How many letters are there in the third chapter of the book volume 3 of "Around the World in 80 Days"?
- Is the word “Lustful” appear in the first chapter?
- What is the sentence beginning at 275th paragraph of Chapter 4?



Read the book and answer questions.



Read all books and answer questions !
Can take more than a life.



Read the superBook and answer questions !!

Vocabulary

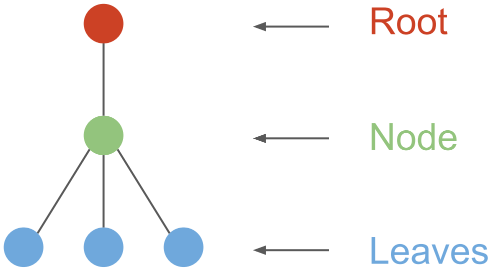
Alphabet, prefix, factor, suffix

Alphabet → $\sum = \{A, C, G, T\}$

Prefix Factor Suffix

CAGCTGACTAGCACGAACT

Root, node, leaves



K-mer

A fragment of k consecutive nucleotides of a word (a sequence from a reference genome as appropriate)

→ A k-mer is a k size factor of a word

Our hypothesis

In rice genome, the number of distinct k-mers (which appear at least once) tends to stabilize from a x number of genomes.

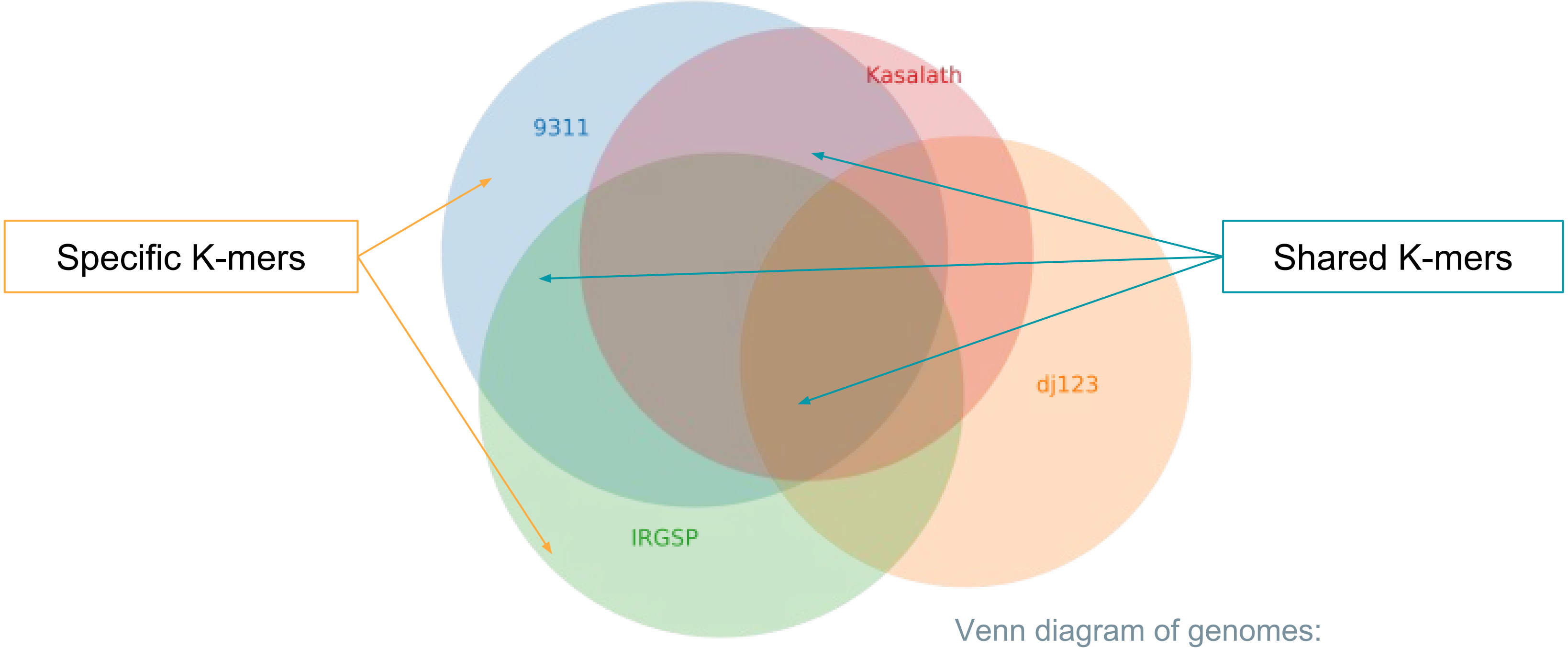
Adding a new genome to an existing index created on 1000 genomes will be equal to adding a reduct set of positions.

To validate this hypothesis:

- 4 genomes + meta-chart (1) → Venn chart
- 8 genomes + k-mers counter (JellyFish) (2)

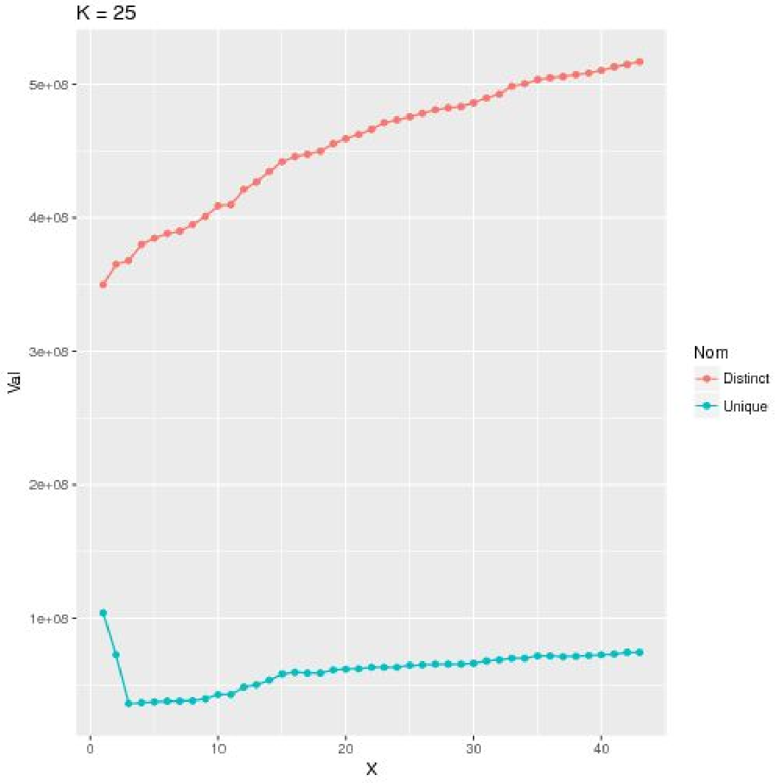
1- https://www.meta-chart.com/   2 - JellyFish : Guillaume Marçais et al

Validation of hypothesis



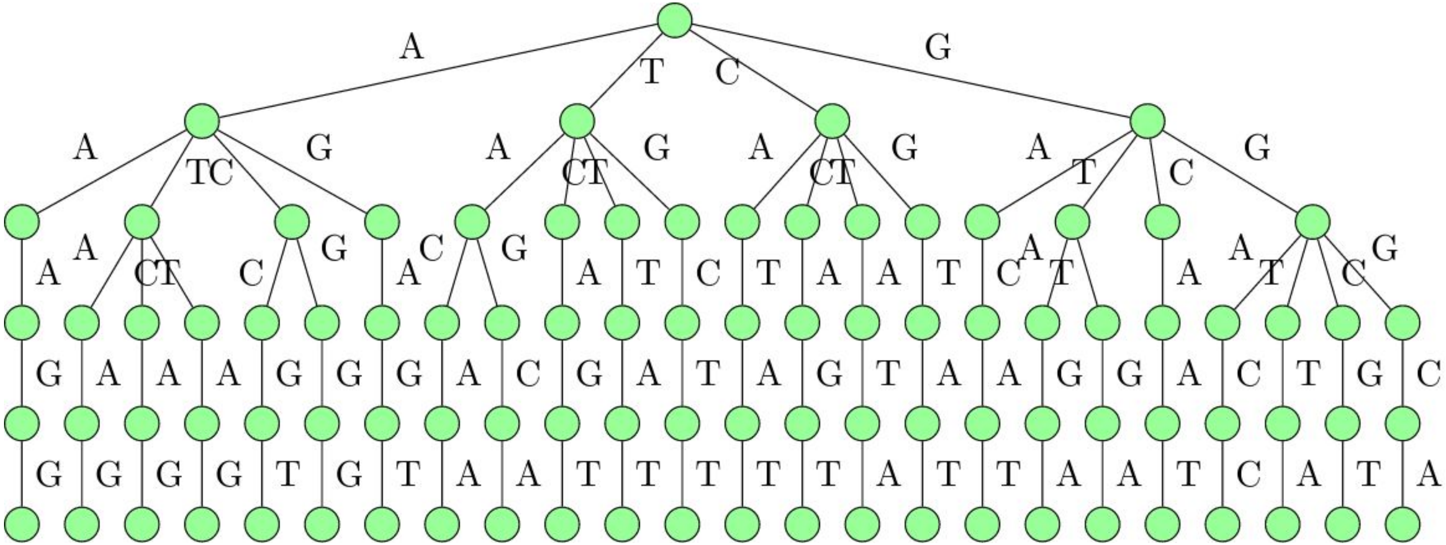
Venn diagram of genomes:
9311, IRGSP, dj123 et Kasalath.

Our study



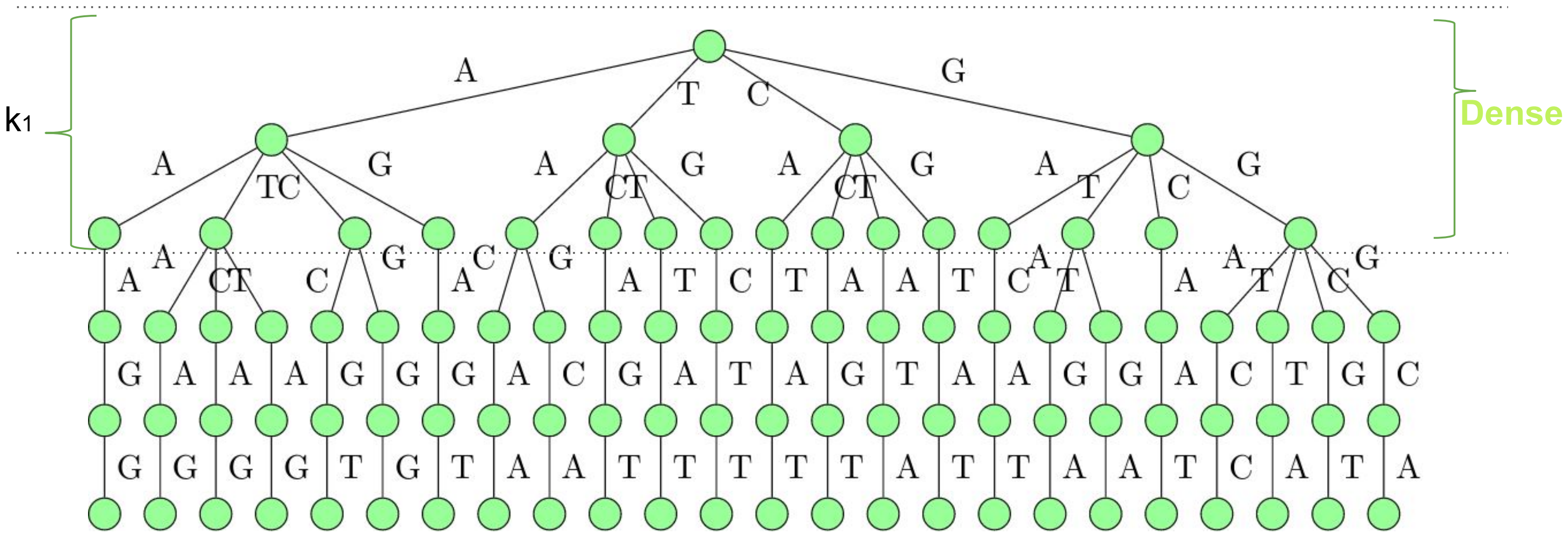
Distinct: Appears at least once
Unique: Appears exactly once

Our approach



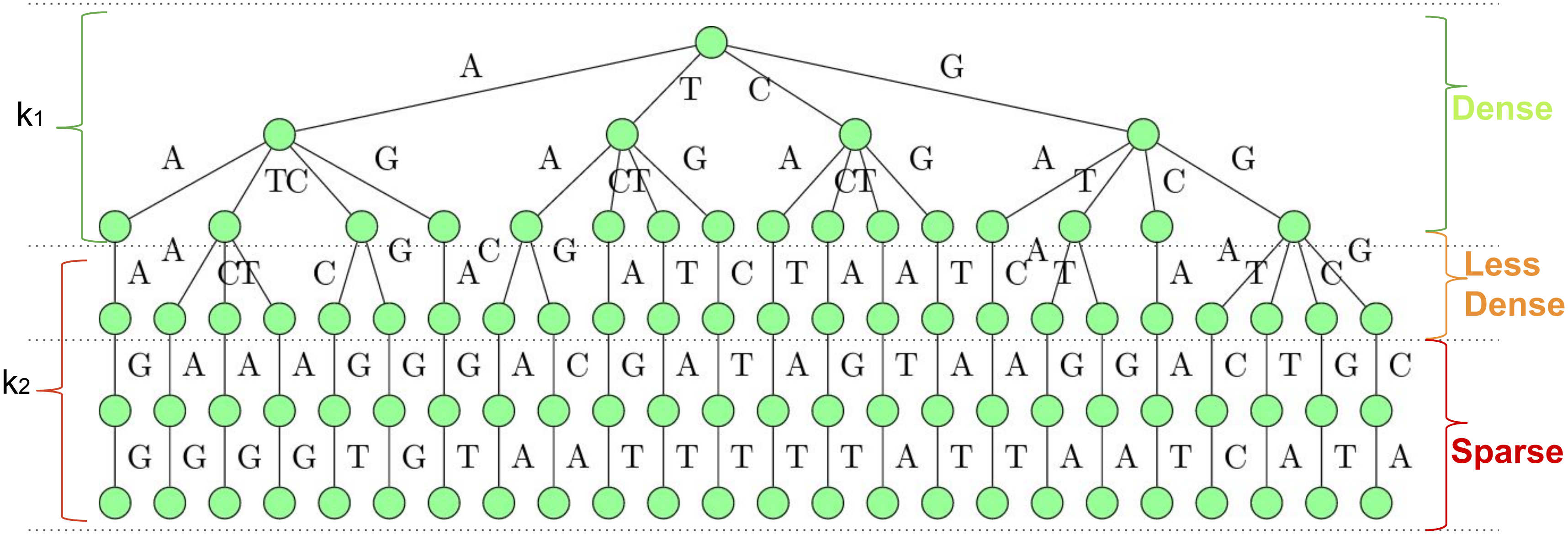
G. Park et al, « Profiles of Tries » janv. 2009.

Our approach



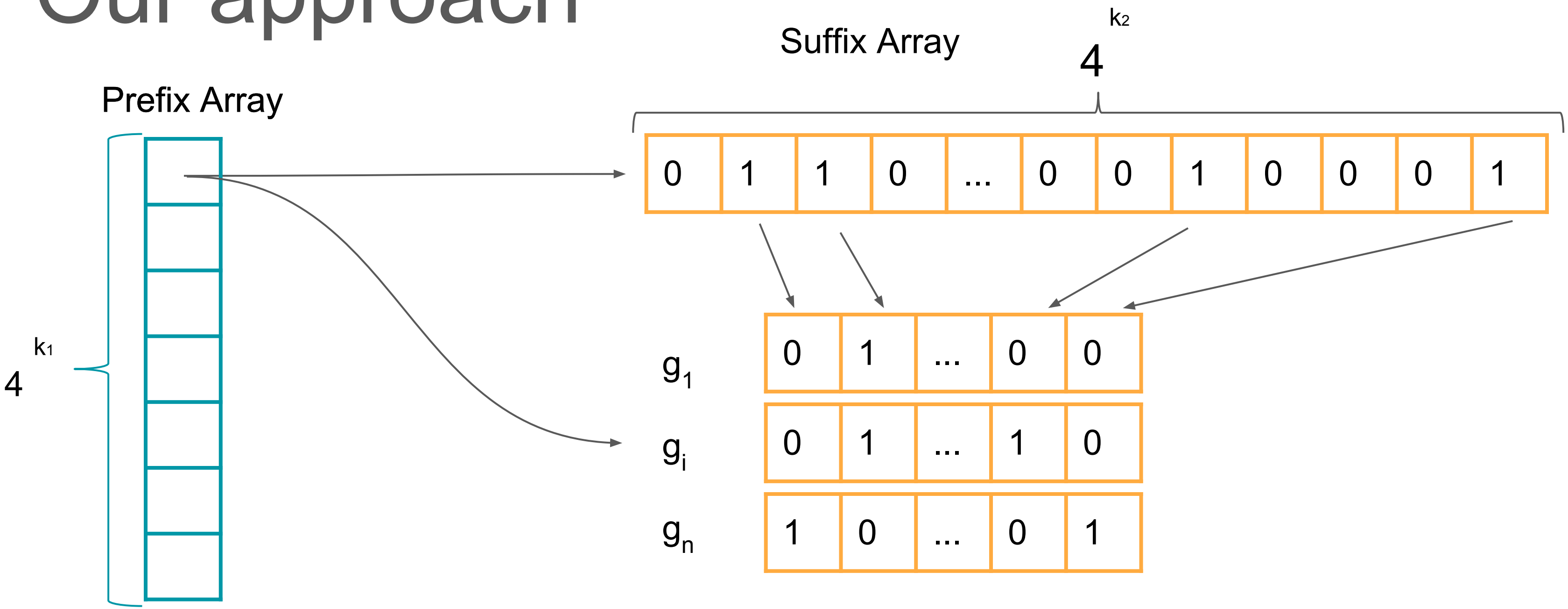
G. Park et al, « Profiles of Tries » janv. 2009.

Our approach



G. Park et al, « Profiles of Tries » janv. 2009.

Our approach



Examples

Alphabet → $\sum = \{A, B\}$
$K = 6 \rightarrow K_1 = 2$ & $K_2 = 4$

g1 = ABBABAABAB
g2 = ABBBAABABB
g3 = AABBBABABA

Let's Create the superBook !

Examples

Alphabet → ∑ = {A, B}
$K = 6 \rightarrow K_1 = 2$ & $K_2 = 4$

g1 = ABBABAABAB
g2 = ABBBAABABB
g3 = AABBBABABA

AAAA, AAAB, AABA, AABB, ABAA, ABAB, ABBA, ABBB, BAAA, BAAB, BABA, BABB, BBAA,BBAB, BBBA, BBBB

Prefix Array (2^2)		Suffix Array
AA	→	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
AB	→	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
BA	→	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
BB	→	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

AA	→	g1 = g2 = g3 =
AB	→	g1 = g2 = g3 =
BA	→	g1 = g2 = g3 =
BB	→	g1 = g2 = g3 =

Examples

Alphabet → ∑ = {A, B}
K = 6 → K_1 = 2 & K_2 = 4

g1 = ABBABAABAB
g2 = ABBBAABABB
g3 = AABBBABABA

AAAA, AAAB, AABA, AABB, ABAA, ABAB, ABBA, ABBB, BAAA, BAAB, BABA, BABB, BBAA,BBAB, BBBA, BBBB

Prefix Array | Suffix Array

2^2

Prefix Array		Suffix Array
AA	→	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
AB	→	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
BA	→	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
BB	→	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

g1 = ABBABAABAB

AA	→	g1 = g2 = g3 =
AB	→	g1 = g2 = g3 =
BA	→	g1 = g2 = g3 =
BB	→	g1 = g2 = g3 =

Examples

AAAA, AAAB, AABA, AABB, ABAA, ABAB, ABBA, ABBB, BAAA, BAAB, BABA, BABB, BBAA,BBAB, BBBA, BBBB

Alphabet → ∑ = {A, B}
$K = 6 \rightarrow K_1 = 2$ & $K_2 = 4$

g1 = ABBABAABAB
g2 = ABBBAABABB
g3 = AABBBABABA

Prefix Array		Suffix Array
AA	→	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
AB	→	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
BA	→	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
BB	→	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

2^2

g1 = ABBABAABAB
ABBABA

AA	→	g1 = g2 = g3 =
AB	→	g1 = g2 = g3 =
BA	→	g1 = g2 = g3 =
BB	→	g1 = g2 = g3 =

Examples

AAAA, AAAB, AABA, AABB, ABAA, ABAB, ABBA, ABBB, BAAA, BAAB, BABA, BABB, BBAA,BBAB, BBBA, BBBB

Alphabet → $\sum$ = {A, B}
K = 6 → K_1 = 2 & K_2 = 4

g1 = ABBABAABAB
g2 = ABBBAABABB
g3 = AABBBABABA

Prefix Array / Suffix Array

2^2

Prefix Array	Suffix Array
AA	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
AB	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
BA	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
BB	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

g1 = ABBABAABAB
ABBABA

ABBABA

AA → g1 = g2 = g3 =

AB → g1 = g2 = g3 =

BA → g1 = g2 = g3 =

BB → g1 = g2 = g3 =

Examples

Alphabet → ∑ = {A, B}
$K = 6 \rightarrow K_1 = 2$ & $K_2 = 4$

g1 = ABBABAABAB
g2 = ABBBAABABB
g3 = AABBBABABA

AAAA, AAAB, AABA, AABB, ABAA, ABAB, ABBA, ABBB, BAAA, BAAB, BABA, BABB, BBAA,BBAB, BBBA, BBBB

Prefix Array		Suffix Array
AA	→	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
AB	→	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
BA	→	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
BB	→	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

2^2

g1 = ABBABAABAB
ABBABA

ABBABA

AA	→	g1 = g2 = g3 =
AB	→	g1 = g2 = g3 =
BA	→	g1 = g2 = g3 =
BB	→	g1 = g2 = g3 =

Examples

AAAA, AAAB, AABA, AABB, ABAA, ABAB, ABBA, ABBB, BAAA, BAAB, BABA, BABB, BBAA,BBAB, BBBA, BBBB

Alphabet → ∑ = {A, B}
$K = 6 \rightarrow K_1 = 2$ & $K_2 = 4$

g1 = ABBABAABAB
g2 = ABBBAABABB
g3 = AABBBABABA

Prefix Array (2^2)		Suffix Array
AA	→	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
AB	→	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
BA	→	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
BB	→	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

g1 = ABBABAABAB
ABBABA

ABBABA

AA	→	g1 = g2 = g3 =
AB	→	g1 = g2 = g3 =
BA	→	g1 = g2 = g3 =
BB	→	g1 = g2 = g3 =

Examples

AAAA, AAAB, AABA, AABB, ABAA, ABAB, ABBA, ABBB, BAAA, BAAB, BABA, BABB, BBAA,BBAB, BBBA, BBBB

Alphabet → ∑ = {A, B}

$K = 6 \rightarrow K_1 = 2$ & $K_2 = 4$

g1 = ABBABAABAB
g2 = ABBBAABABB
g3 = AABBBABABA

Prefix Array

Suffix Array

2^2

Prefix Array	Suffix Array
AA	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
AB	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
BA	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
BB	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

g1 = ABBABAABAB
ABBABA

ABBABA

AA	g1 = g2 = g3 =
AB	g1 = g2 = g3 =
BA	g1 = g2 = g3 =
BB	g1 = g2 = g3 =

Examples

AAAA, AAAB, AABA, AABB, ABAA, ABAB, ABBA, ABBB, BAAA, BAAB, BABA, BABB, BBAA,BBAB, BBBA, BBBB

Alphabet → ∑ = {A, B}
$K = 6 \rightarrow K_1 = 2$ & $K_2 = 4$

g1 = ABBABAABAB
g2 = ABBBAABABB
g3 = AABBBABABA

Prefix Array | Suffix Array

2^2

Prefix Array		Suffix Array
AA	→	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
AB	→	0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0
BA	→	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
BB	→	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

g1 = ABBABAABAB
ABBABA

ABBABA

AA	→	g1 = g2 = g3 =
AB	→	g1 = g2 = g3 =
BA	→	g1 = g2 = g3 =
BB	→	g1 = g2 = g3 =

Examples

Alphabet → ∑ = {A, B}
K = 6 → K_1 = 2 & K_2 = 4

g1 = ABBABAABAB
g2 = ABBBAABABB
g3 = AABBBABABA

AAAA, AAAB, AABA, AABB, ABAA, ABAB, ABBA, ABBB, BAAA, BAAB, BABA, BABB, BBAA,BBAB, BBBA, BBBB

Prefix Array | Suffix Array

2^2

AA → 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
AB → 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0
BA → 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
BB → 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

g1 = ABBABAABAB
ABBABA

ABBABA

AA → g1 = g2 = g3 =
AB → g1 = g2 = g3 =
BA → g1 = g2 = g3 =
BB → g1 = g2 = g3 =

Examples

Alphabet → ∑ = {A, B}
K = 6 → K_1 = 2 & K_2 = 4

g1 = ABBABAABAB
g2 = ABBBAABABB
g3 = AABBBABABA

AAAA, AAAB, AABA, AABB, ABAA, ABAB, ABBA, ABBB, BAAA, BAAB, BABA, BABB, BBAA,BBAB, BBBA, BBBB

Prefix Array (2^2)	Suffix Array
AA	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
AB	0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0
BA	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
BB	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

g1 = ABBABAABAB
ABBABA

ABBABA

Prefix	
AA	g1 = g2 = g3 =
AB	g1 = 1 g2 = 0 g3 = 0
BA	g1 = g2 = g3 =
BB	g1 = g2 = g3 =

Examples

Alphabet → ∑ = {A, B}
K = 6 → K_1 = 2 & K_2 = 4

g1 = ABBABAABAB
g2 = ABBBAABABB
g3 = AABBBABABA

AAAA, AAAB, AABA, AABB, ABAA, ABAB, ABBA, ABBB, BAAA, BAAB, BABA, BABB, BBAA,BBAB, BBBA, BBBB

Prefix Array	Suffix Array
AA	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
AB	0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0
BA	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
BB	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

2^2

g1 = ABBABAABAB
ABBABA

AA → g1 =
g2 =
g3 =

AB → g1 = 1
g2 = 0
g3 = 0

BA → g1 =
g2 =
g3 =

BB → g1 =
g2 =
g3 =

Examples

Alphabet → ∑ = {A, B}
K = 6 → K_1 = 2 & K_2 = 4

g1 = ABBABAABAB
g2 = ABBBAABABB
g3 = AABBBABABA

AAAA, AAAB, AABA, AABB, ABAA, ABAB, ABBA, ABBB, BAAA, BAAB, BABA, BABB, BBAA,BBAB, BBBA, BBBB

Prefix Array | Suffix Array

2^2

Prefix Array	Suffix Array
AA	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
AB	0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0
BA	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
BB	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

g1 = ABBABAABAB
ABBABA
BBABAA
BABAAB
ABAABA
BAABAB

AA	g1 = g2 = g3 =
AB	g1 = 1 g2 = 0 g3 = 0
BA	g1 = g2 = g3 =
BB	g1 = g2 = g3 =

Examples

Alphabet → ∑ = {A, B}
$K = 6 \rightarrow K_1 = 2$ & $K_2 = 4$

g1 = ABBABAABAB
g2 = ABBBAABABB
g3 = AABBBABABA

AAAA, AAAB, AABA, AABB, ABAA, ABAB, ABBA, ABBB, BAAA, BAAB, BABA, BABB, BBAA,BBAB, BBBA, BBBB

Prefix Array	Suffix Array
AA	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
AB	0 0 1 0 0 0 0 0 0 0 1 0 0 0 0 0
BA	0 0 0 0 0 1 0 0 0 1 0 0 0 0 0 0
BB	0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0

2^2

g1 = ABBABAABAB
ABBABA
BBABAA
BABAAB
ABAABA
BAABAB

AA → g1 = g2 = g3 =

AB → g1 = 11 g2 = 00 g3 = 00

BA → g1 = 11 g2 = 00 g3 = 00

BB → g1 = g2 = g3 =

Examples

Alphabet → ∑ = {A, B}
$K = 6 \rightarrow K_1 = 2$ & $K_2 = 4$

g1 = ABBABAABAB
g2 = ABBBAABABB
g3 = AABBBABABA

AAAA, AAAB, AABA, AABB, ABAA, ABAB, ABBA, ABBB, BAAA, BAAB, BABA, BABB, BBAA,BBAB, BBBA, BBBB

2^2

Prefix Array	Suffix Array
AA	0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0
AB	0 0 1 0 0 0 0 0 0 0 1 0 1 1 0 0
BA	0 0 0 0 0 1 0 0 0 1 1 0 0 0 0 0
BB	0 0 1 0 1 1 0 0 0 1 1 0 0 0 0 0

AA → g1 = 00, g2 = 10, g3 = 01

AB → g1 = 1100, g2 = 0010, g3 = 0001

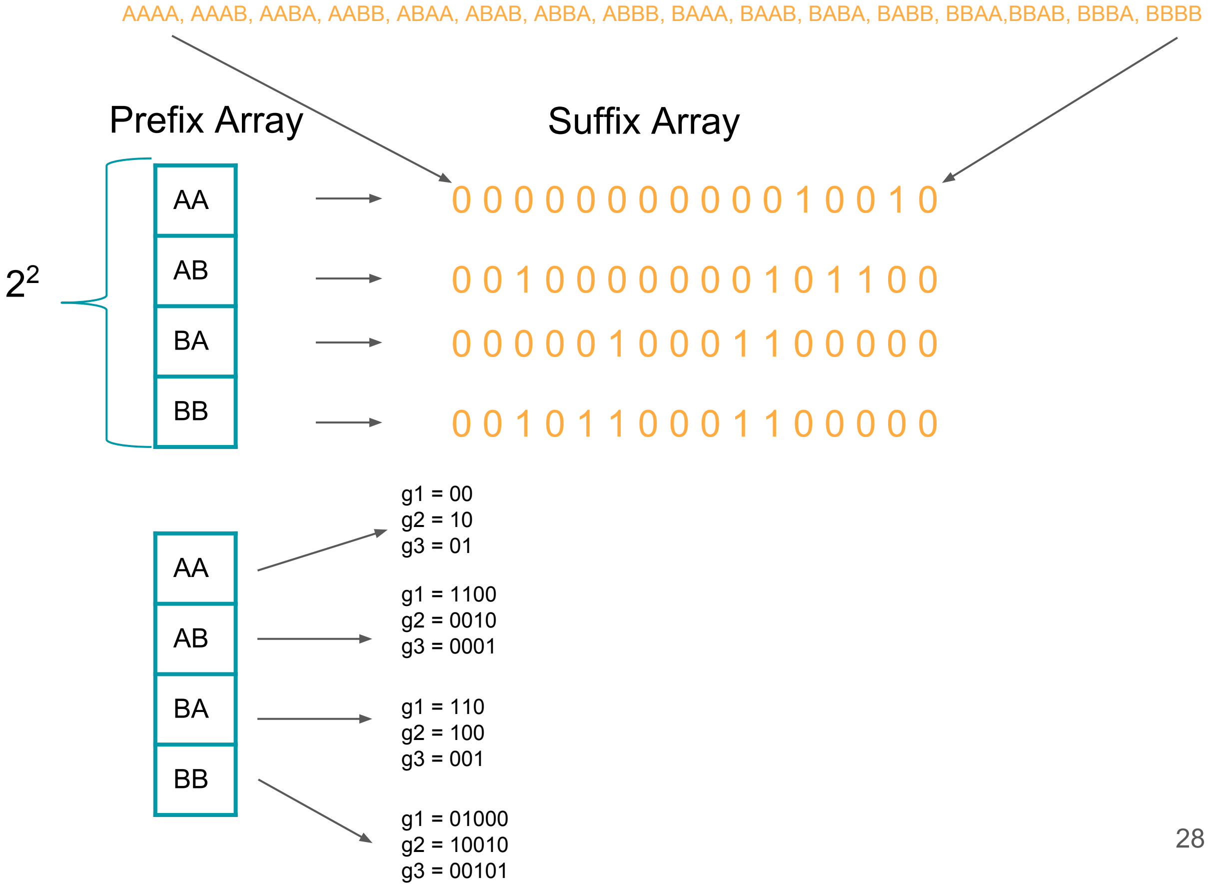
BA → g1 = 110, g2 = 100, g3 = 001

BB → g1 = 01000, g2 = 10010, g3 = 00101

Examples

Alphabet → $\Sigma = \{A, B\}$
$K = 6 \rightarrow K_1 = 2$ & $K_2 = 4$

g1 = ABBABAABAB
g2 = ABBBAABABB
g3 = AABBBABABA



$\text{Rank}_x(i)$:
Rank return the number of elements x in the range [0,i].

$\text{Select}_x(i)$:
Select is the inverse operation to rank; it answers the question "at which position is the i^{th} set bit?"

Examples

AAAA, AAAB, AABA, AABB, ABAA, ABAB, ABBA, ABBB, BAAA, BAAB, BABA, BABB, BBAA,BBAB, BBBA, BBBB

Is the word BBABAA exist ?

BBABAA

Prefix Array	Suffix Array
AA	0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0
AB	0 0 1 0 0 0 0 0 0 0 1 0 1 1 0 0
BA	0 0 0 0 0 1 0 0 0 1 1 0 0 0 0 0
BB	0 0 1 0 1 1 0 0 0 1 1 0 0 0 0 0

2^2

AA	g1 = 00 g2 = 10 g3 = 01
AB	g1 = 1100 g2 = 0010 g3 = 0001
BA	g1 = 110 g2 = 100 g3 = 001
BB	g1 = 01000 g2 = 10010 g3 = 00101

Examples

AAAA, AAAB, AABA, AABB, ABAA, ABAB, ABBA, ABBB, BAAA, BAAB, BABA, BABB, BBAA,BBAB, BBBA, BBBB

Is the word BBABAA exist ?

BBABAA

2^2

Prefix Array	Suffix Array
AA	0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0
AB	0 0 1 0 0 0 0 0 0 0 0 1 0 1 1 0 0
BA	0 0 0 0 0 1 0 0 0 1 1 0 0 0 0 0
BB	0 0 1 0 1 1 0 0 0 1 1 0 0 0 0 0

AA	g1 = 00 g2 = 10 g3 = 01
AB	g1 = 1100 g2 = 0010 g3 = 0001
BA	g1 = 110 g2 = 100 g3 = 001
BB	g1 = 01000 g2 = 10010 g3 = 00101

Examples

AAAA, AAAB, AABA, AABB, ABAA, ABAB, ABBA, ABBB, BAAA, BAAB, BABA, BABB, BBAA,BBAB, BBBA, BBBB

Is the word BBABAA exist ?

BBABAA

Prefix Array	Suffix Array
AA	0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0
AB	0 0 1 0 0 0 0 0 0 0 0 1 0 1 1 0 0
BA	0 0 0 0 0 1 0 0 0 1 1 0 0 0 0 0
BB	0 0 1 0 1 1 0 0 0 1 1 0 0 0 0 0

2^2

AA	g1 = 00 g2 = 10 g3 = 01
AB	g1 = 1100 g2 = 0010 g3 = 0001
BA	g1 = 110 g2 = 100 g3 = 001
BB	g1 = 01000 g2 = 10010 g3 = 00101

Examples

AAAA, AAAB, AABA, AABB, ABAA, ABAB, ABBA, ABBB, BAAA, BAAB, BABA, BABB, BBAA,BBAB, BBBA, BBBB

Is the word BBABAA exist ?

BBABAA

In which genomes ?
$Rank_1(ABAA) = 2$

Prefix Array | Suffix Array

2^2

Prefix Array	Suffix Array
AA	0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0
AB	0 0 1 0 0 0 0 0 0 0 1 0 1 1 0 0
BA	0 0 0 0 0 1 0 0 0 1 1 0 0 0 0 0
BB	0 0 1 0 1 1 0 0 0 1 1 0 0 0 0 0

AA	g1 = 00 g2 = 10 g3 = 01
AB	g1 = 1100 g2 = 0010 g3 = 0001
BA	g1 = 110 g2 = 100 g3 = 001
BB	g1 = 01000 g2 = 10010 g3 = 00101

Examples

AAAA, AAAB, AABA, AABB, ABAA, ABAB, ABBA, ABBB, BAAA, BAAB, BABA, BABB, BBAA,BBAB, BBBA, BBBB

Is the word BBABAA exist ?

BBABAA

In which genomes ?
$\mathrm{Rank}_1(\mathrm{ABAA}) = 2$

2^2

Prefix Array	Suffix Array
AA	0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0
AB	0 0 1 0 0 0 0 0 0 0 1 0 1 1 0 0
BA	0 0 0 0 0 1 0 0 0 1 1 0 0 0 0 0
BB	0 0 1 0 1 1 0 0 0 1 1 0 0 0 0 0

AA → g1 = 00, g2 = 10, g3 = 01

AB → g1 = 1100, g2 = 0010, g3 = 0001

BA → g1 = 110, g2 = 100, g3 = 001

BB → g1 = 01000, g2 = 10010, g3 = 00101

Examples

AAAA, AAAB, AABA, AABB, ABAA, ABAB, ABBA, ABBB, BAAA, BAAB, BABA, BABB, BBAA,BBAB, BBBA, BBBB

Is the word BBABAA exist ?

BBABAA

In which genomes ?
$Rank_1(ABAA) = 2$
→ g1 only!

Reminder :
g1 = ABBABAABAB
g2 = ABBBAABABB
g3 = AABBBABABA

Prefix Array		Suffix Array
AA	→	0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0
AB	→	0 0 1 0 0 0 0 0 0 0 1 0 1 1 0 0
BA	→	0 0 0 0 0 1 0 0 0 1 1 0 0 0 0 0
BB	→	0 0 1 0 1 1 0 0 0 1 1 0 0 0 0 0

2^2

AA → g1 = 00, g2 = 10, g3 = 01

AB → g1 = 1100, g2 = 0010, g3 = 0001

BA → g1 = 110, g2 = 100, g3 = 001

BB → g1 = 01000, g2 = 10010, g3 = 00101

RRR

RRR was first proposed by Raman et al [1]

→ $O(1)$ time binary rank queries

→ $N\,H_0(S) + o(N)$ $H_0(S)$ is the zeroth-order empirical entropy of S

[1] R. Raman, V. Raman, and S. R. Satti. Succinct indexable dictionaries with applications to encoding k-ary trees, prefix sums and multisets. ACM Transactions on Algorithms, 3(4), 2007.

RRR

0101 0110 1011 0000 1001 1001 1011 0101 1011

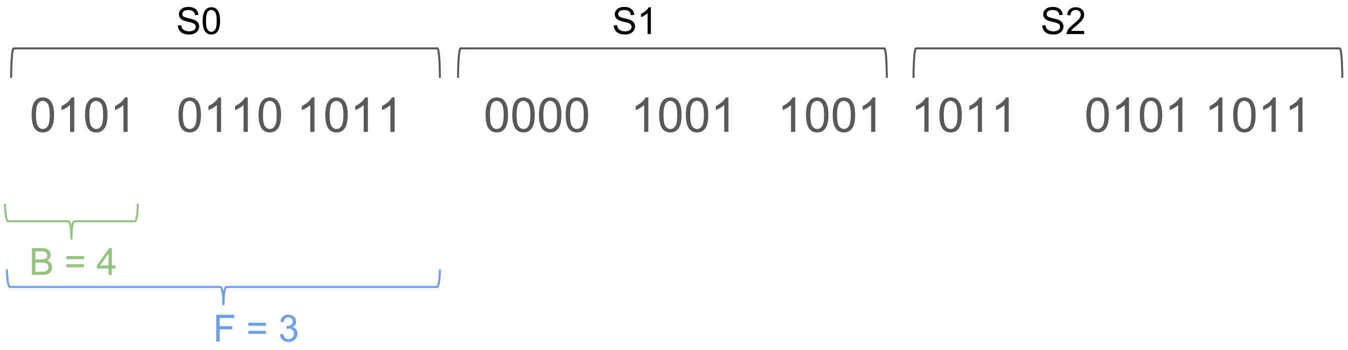
[1] R. Raman, V. Raman, and S. R. Satti. Succinct indexable dictionaries with applications to encoding k-ary trees, prefix sums and multisets. ACM Transactions on Algorithms, 3(4), 2007.

RRR

$$\overbrace{0101 \quad 0110\ 1011}^{S0} \quad \overbrace{0000 \quad 1001 \quad 1001}^{S1} \quad \overbrace{1011 \quad 0101\ 1011}^{S2}$$

[1] R. Raman, V. Raman, and S. R. Satti. Succinct indexable dictionaries with applications to encoding k-ary trees, prefix sums and multisets. ACM Transactions on Algorithms, 3(4), 2007.

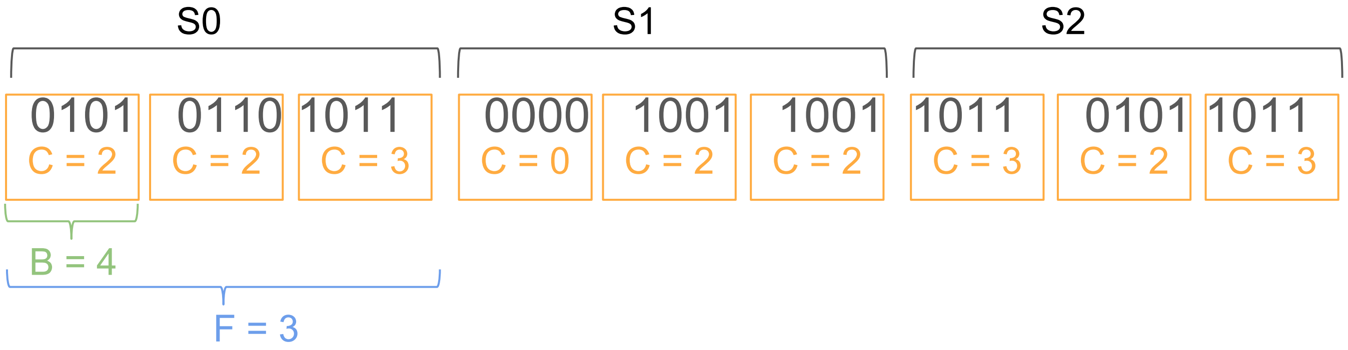
RRR



B : Size of the blocks = Numbers of 1 and 0 in the block
F : Superblock factor
C : Class number = Numbers of 1 in the block b
O : Offset = Index into the table

[1] R. Raman, V. Raman, and S. R. Satti. Succinct indexable dictionaries with applications to encoding k-ary trees, prefix sums and multisets. ACM Transactions on Algorithms, 3(4), 2007.

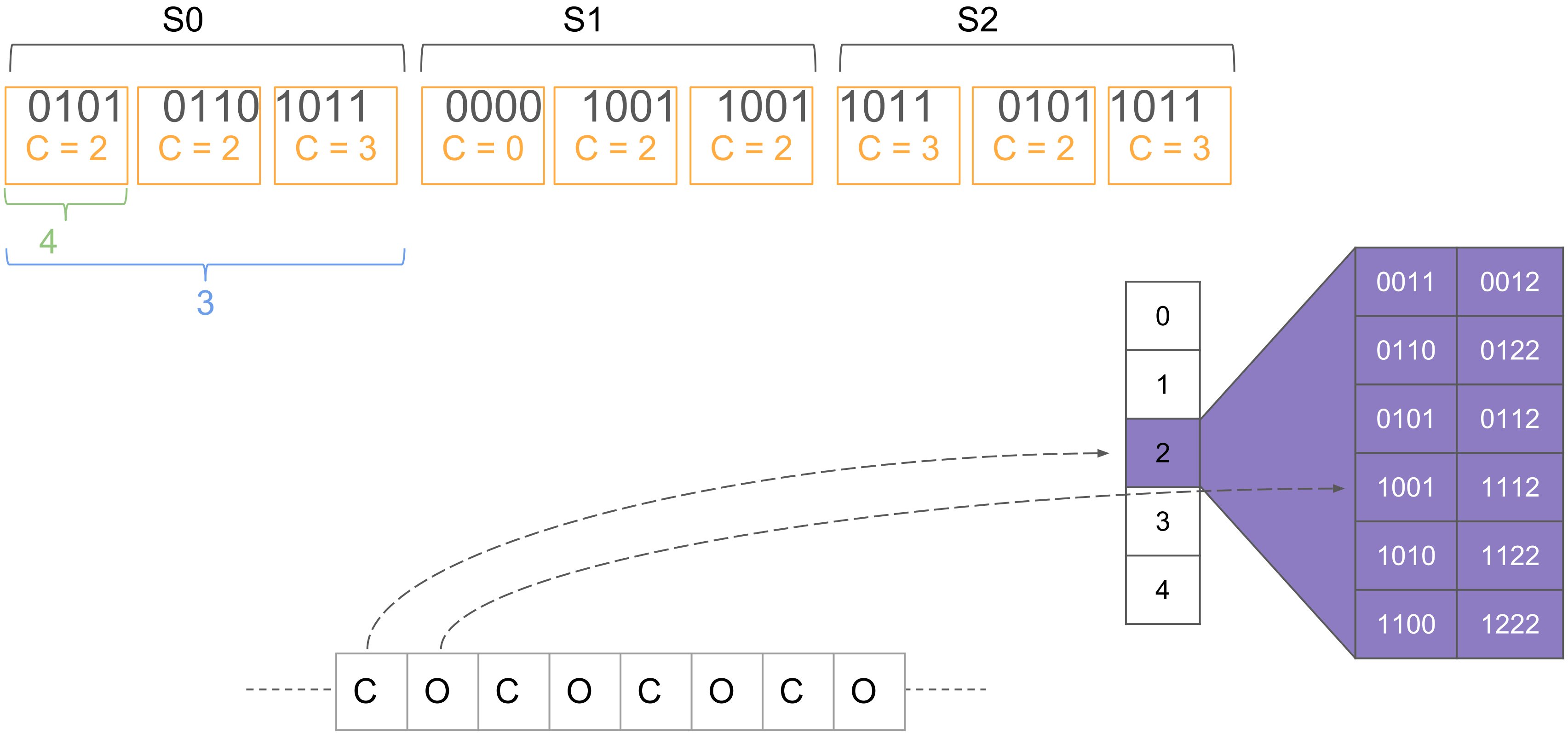
RRR



B : Size of the blocks = Numbers of 1 and 0 in the block
F : Superblock factor
C : Class number = Numbers of 1 in the block b
O : Offset = Index into the table

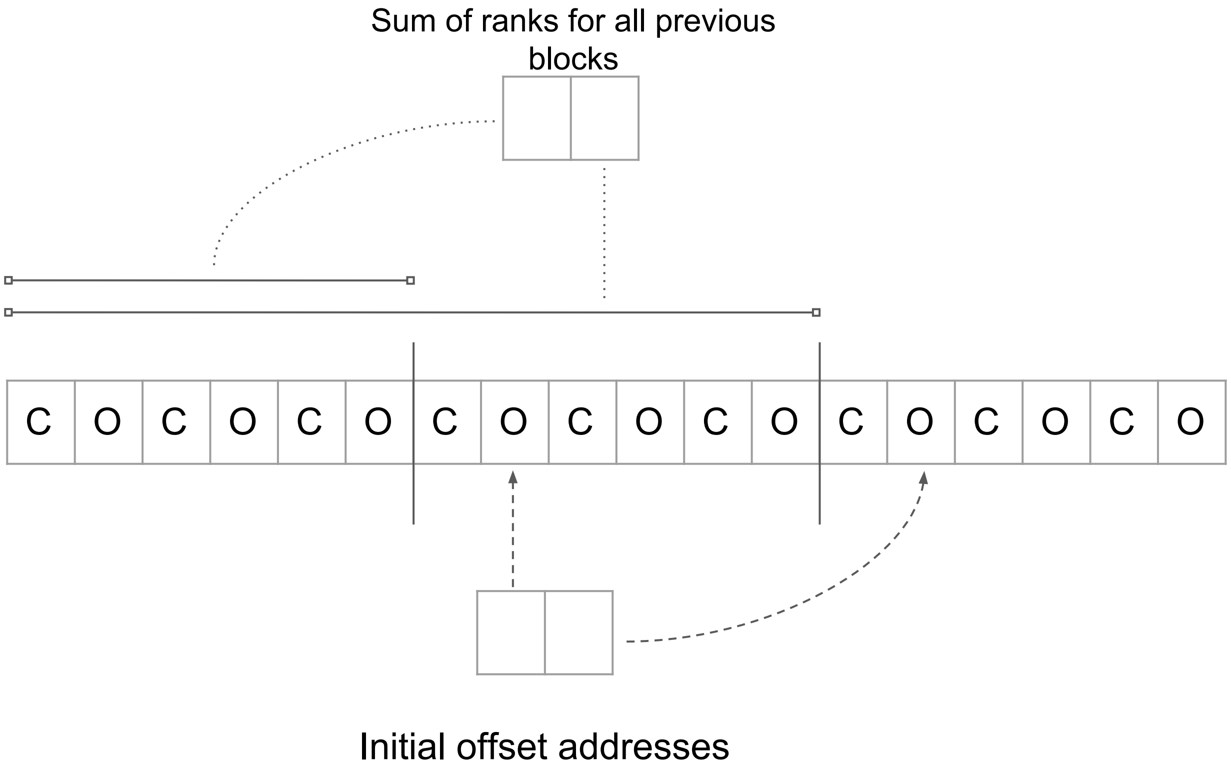
[1] R. Raman, V. Raman, and S. R. Satti. Succinct indexable dictionaries with applications to encoding k-ary trees, prefix sums and multisets. ACM Transactions on Algorithms, 3(4), 2007.

RRR



[1] R. Raman, V. Raman, and S. R. Satti. Succinct indexable dictionaries with applications to encoding k-ary trees, prefix sums and multisets. ACM Transactions on Algorithms, 3(4), 2007.

RRR



http://alexbowe.com/rrr/

RUBIKS :RRR Update for Bit Indexing in K-mer Structure

0000 0100 0000 1000 0000 0100 0000

- Prefixes : 4^8
- Kmers : 10^9
  → 15258 “1”
- Suffixes: 4^{20}

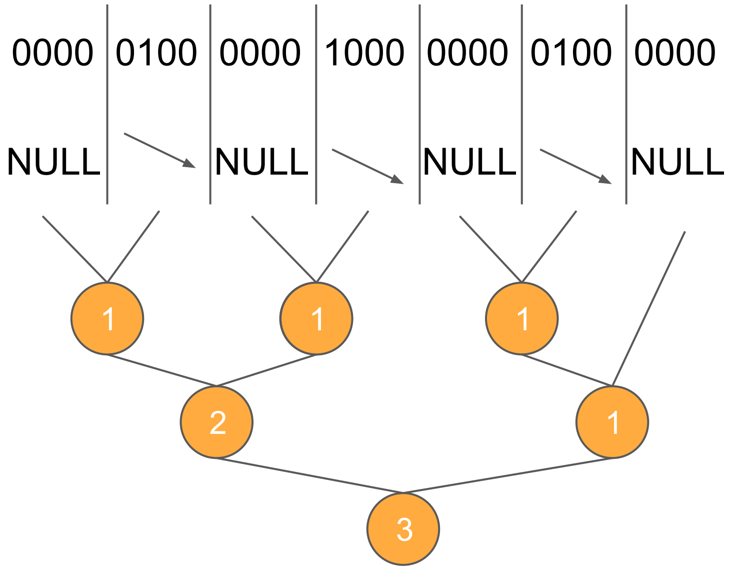
RUBIKS :RRR Update for Bit Indexing in K-mer Structure

0000	0100	0000	1000	0000	0100	0000
NULL		NULL		NULL		NULL

- Prefixes : 4^{8}
- Kmers : 10^{9}
  → 15258 “1”
- Suffixes: 4^{20}

Pointeur vers une RRR

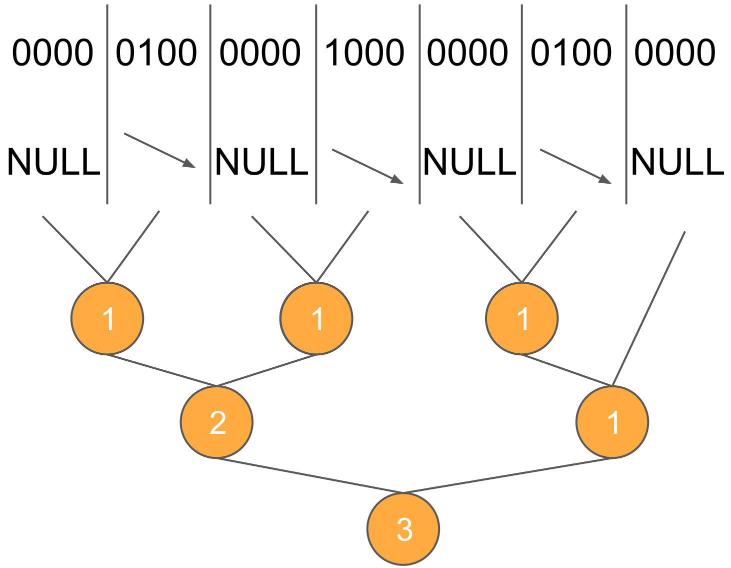
RUBIKS :RRR Update for Bit Indexing in K-mer Structure



- Prefixes : 4^8
- Kmers : 10^9
  → 15258 “1”
- Suffixes: 4^{20}

Pointeur vers une RRR

RUBIKS :RRR Update for Bit Indexing in K-mer Structure



- Prefixes : 4^8
- Kmers : 10^9
  → 15258 “1”
- Suffixes: 4^{20}

Pointeur vers une RRR

We still can do a Rank and Select !

Our outlook

How to answer the following questions:

Is the word "ATATAAGATTACA" present in the first chromosome of genomes 644?

Which sequences are common to the kasalath and 9311 genomes?

SuperBook

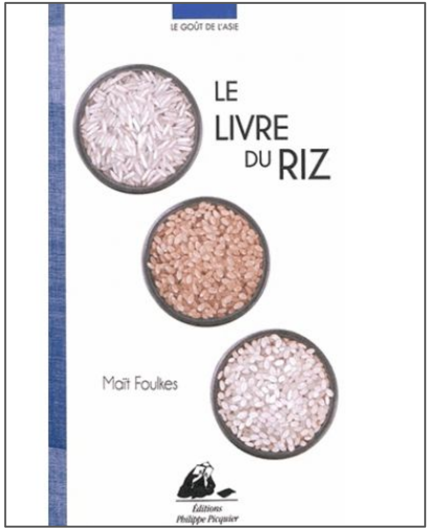
Read the SuperBook and answer questions

Index of 3000 genomes

→ Tools based on this index
→ Integrate these tools into the GenomeHarvest project
→ Create tools to make these index structure easy to use

Thank you for your attention.

Do you have questions ?




That's all Folks!

Annexes

If searching for a word takes 1 sec per line:
→ Search for a word in a book of 500 000 000 lines:
500,000,000 sec: 5,787 Days

→ Search for a word in a dictionary of 500 000 000 lines:
Log (500,000,000) sec: 28,897 sec

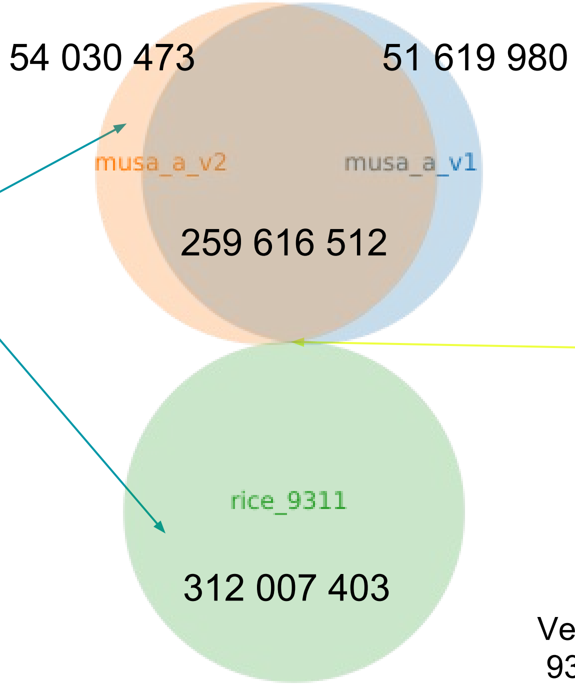
Results

(1) [SDSL Lite]

K_2	compression	b	rrrb $\log \frac{n}{m}$
14		32 Mo	2 Mo
15		128 Mo	4 Mo
16		512 Mo	32 Mo
17		2 Go	128 Mo

Simon and Beller, et al.From Theory to Practice: Plug and Play with Succinct Data Structures, 2014

Validation of hypothesis



Specific K-mers

Shared K-mers

207 353
22 854
21 109

Venn diagram of genomes:
9311, musa_v1, musa_v2

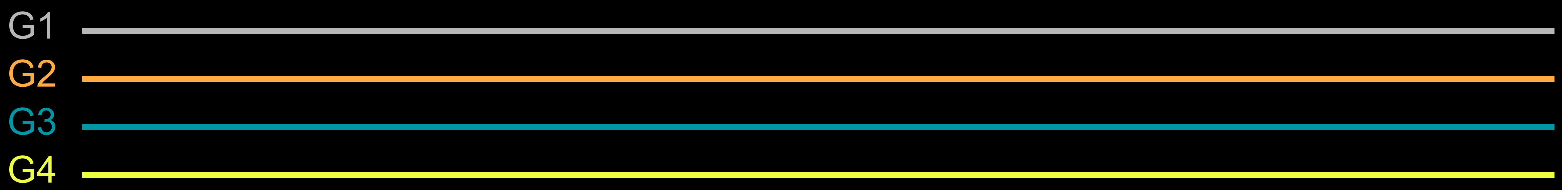
```
#ifdef __aquoicasert
```

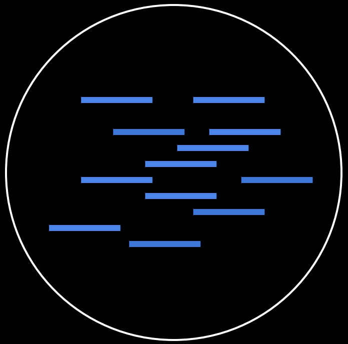
G1

G2

G3

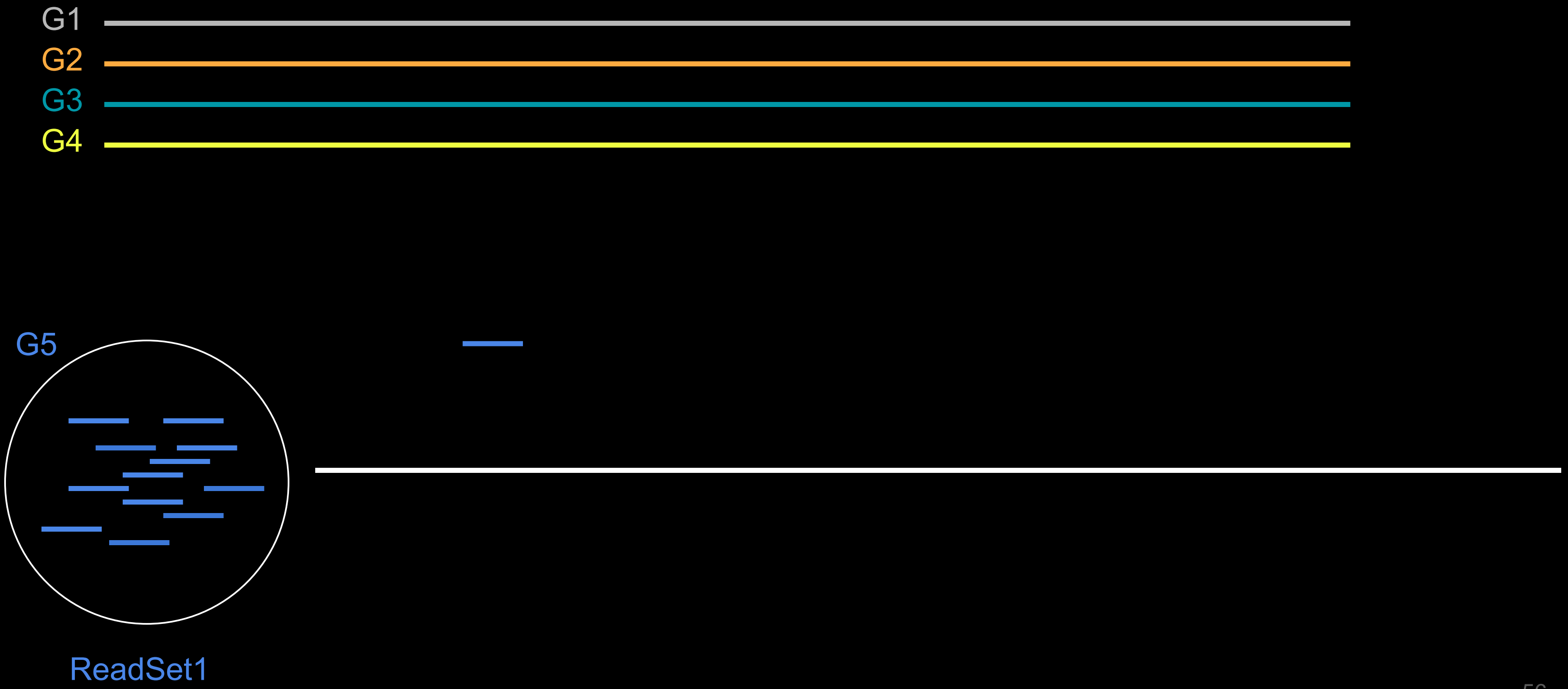
G4

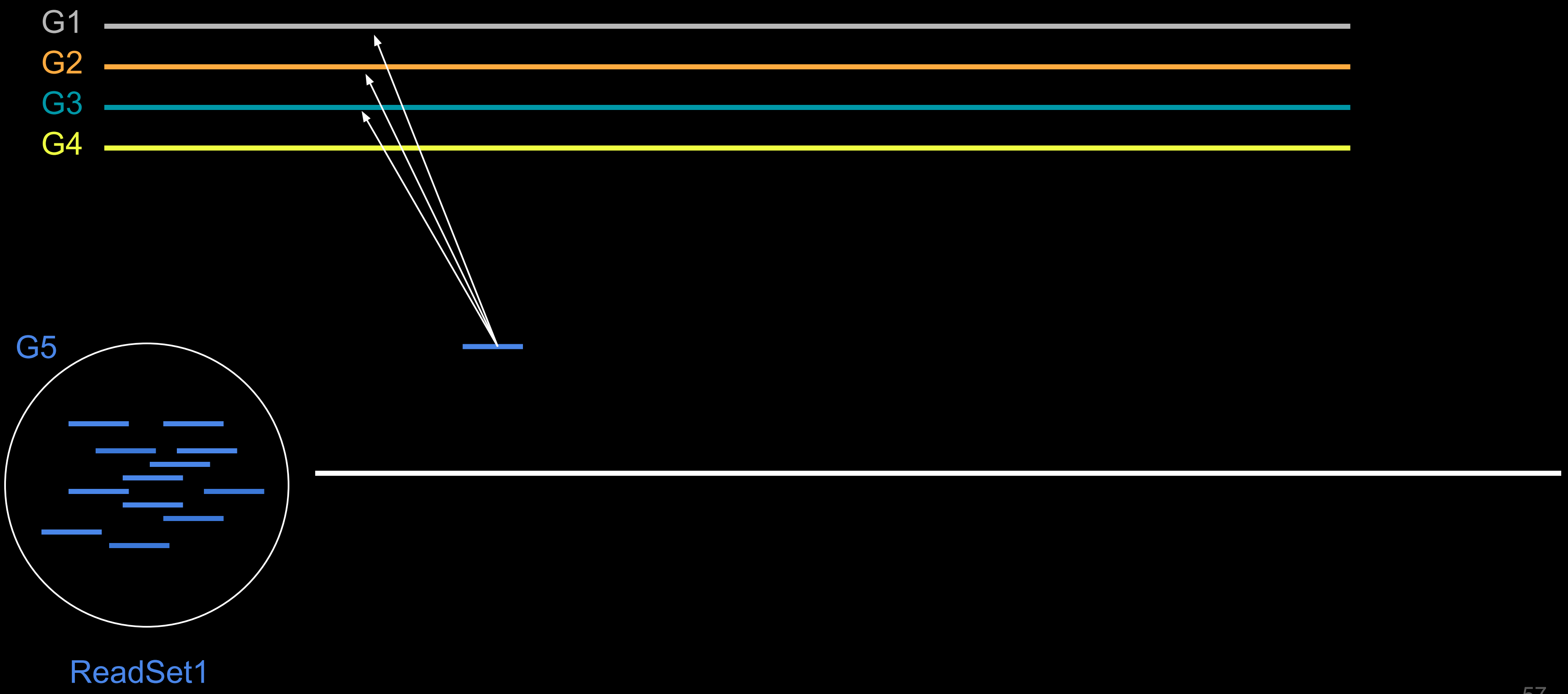


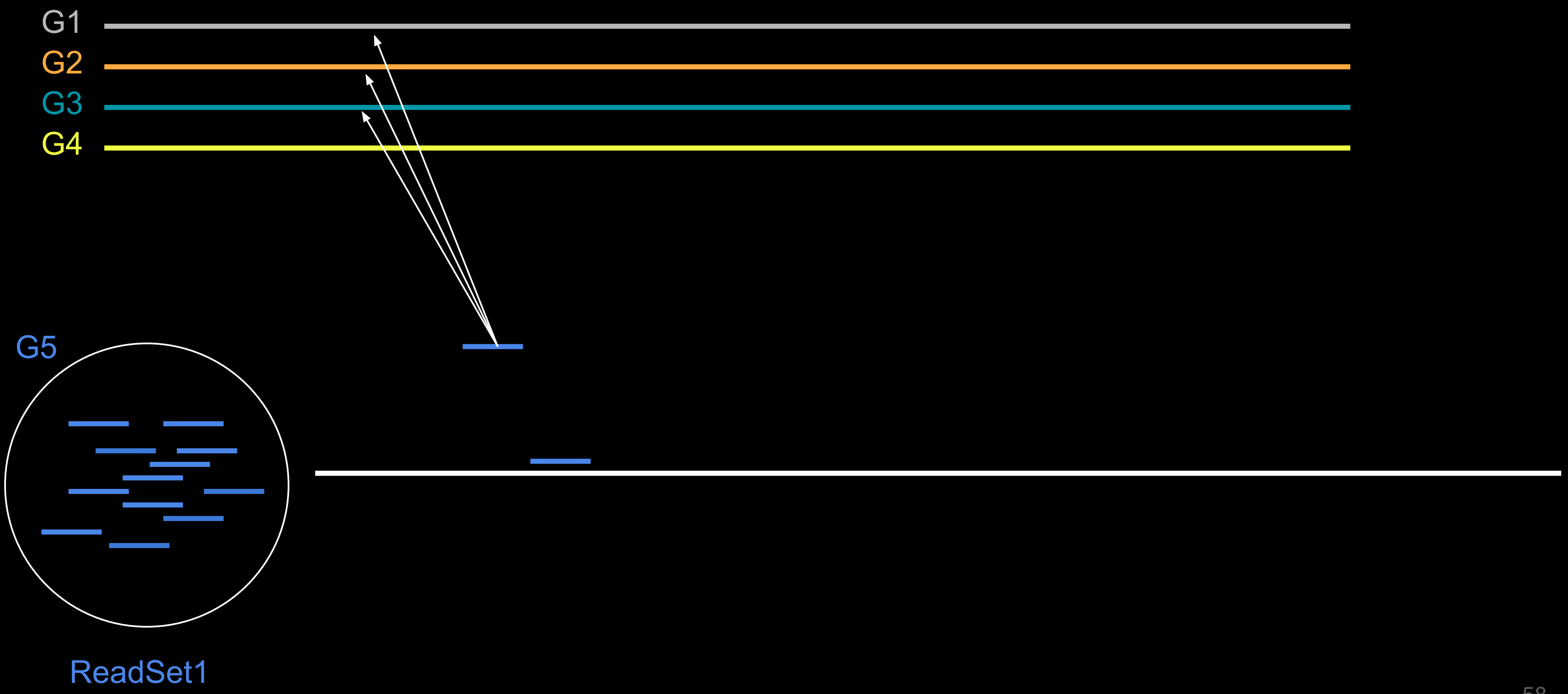


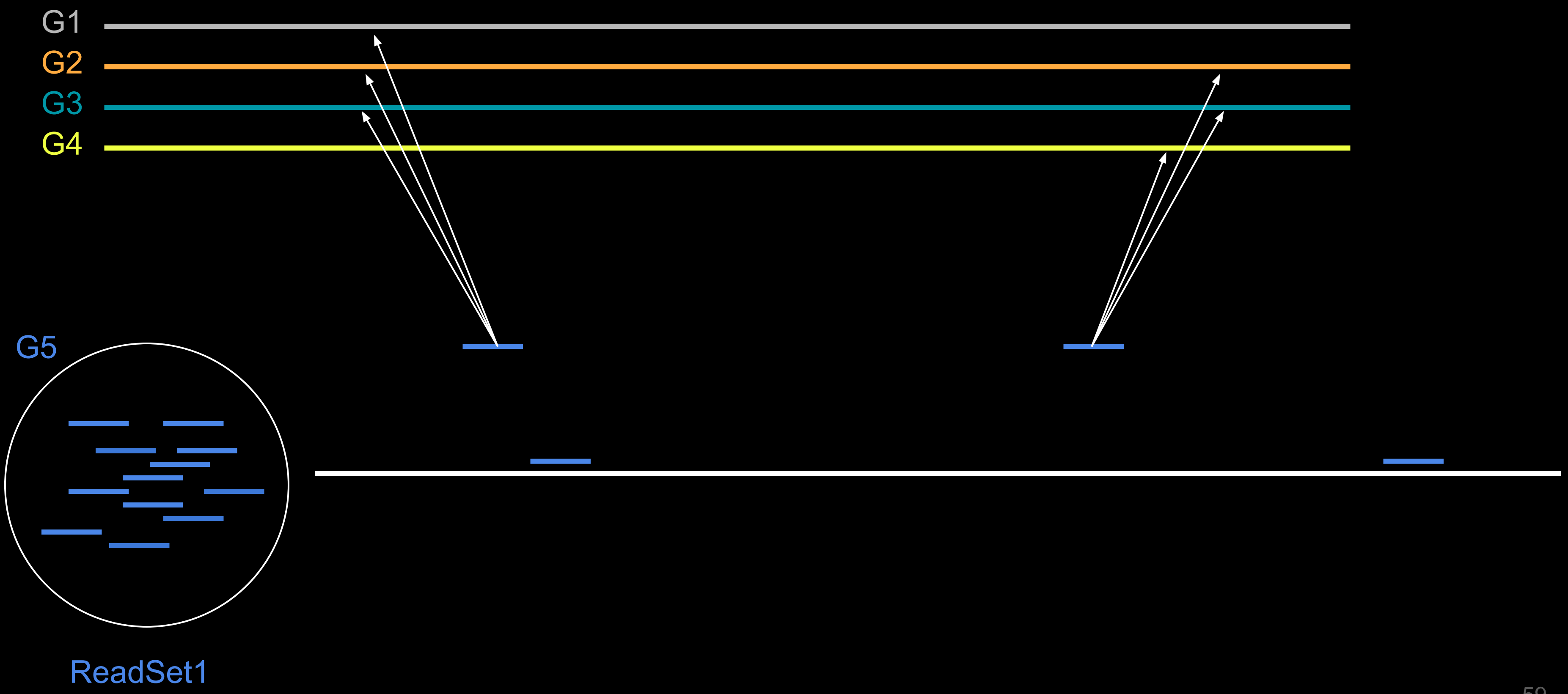
ReadSet1

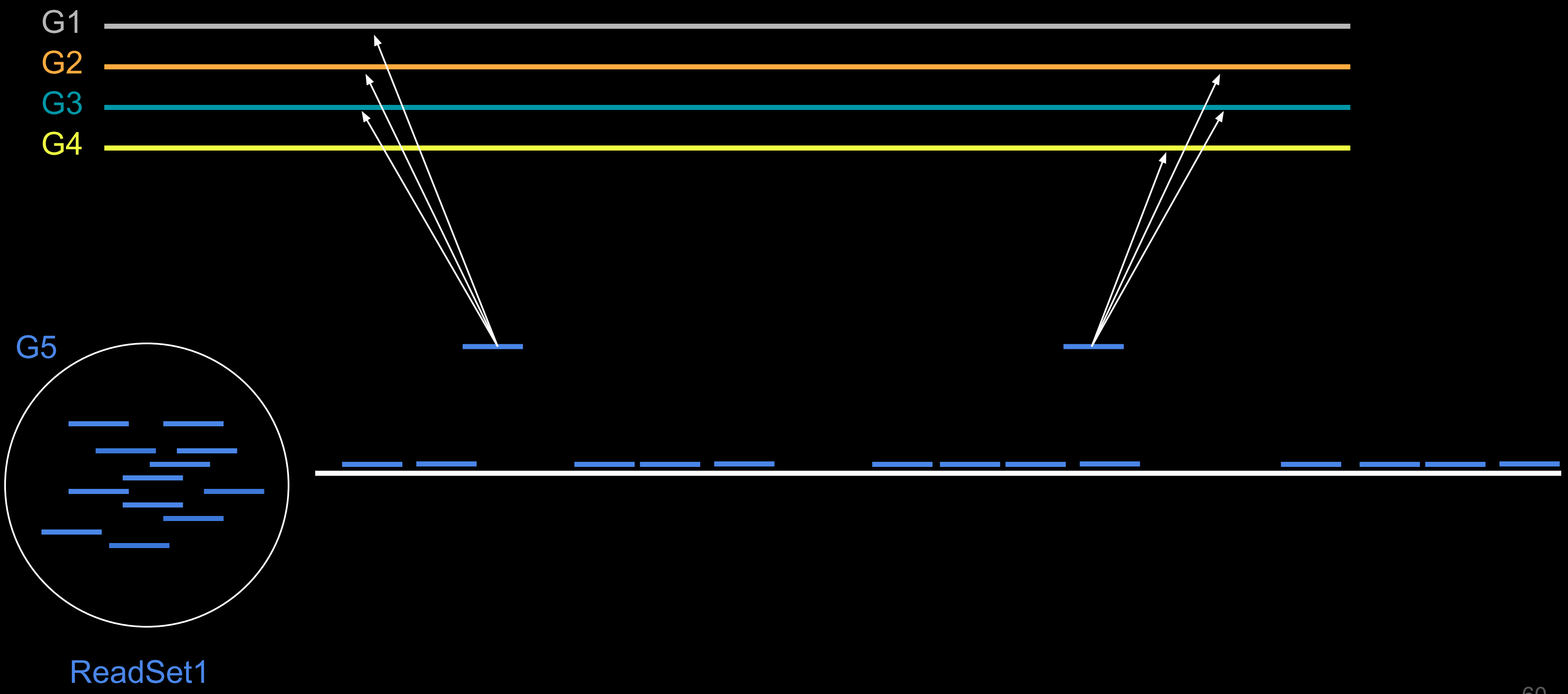

G1
G2
G3
G4
G5
ReadSet1


G1
G2
G3
G4
G5
ReadSet1

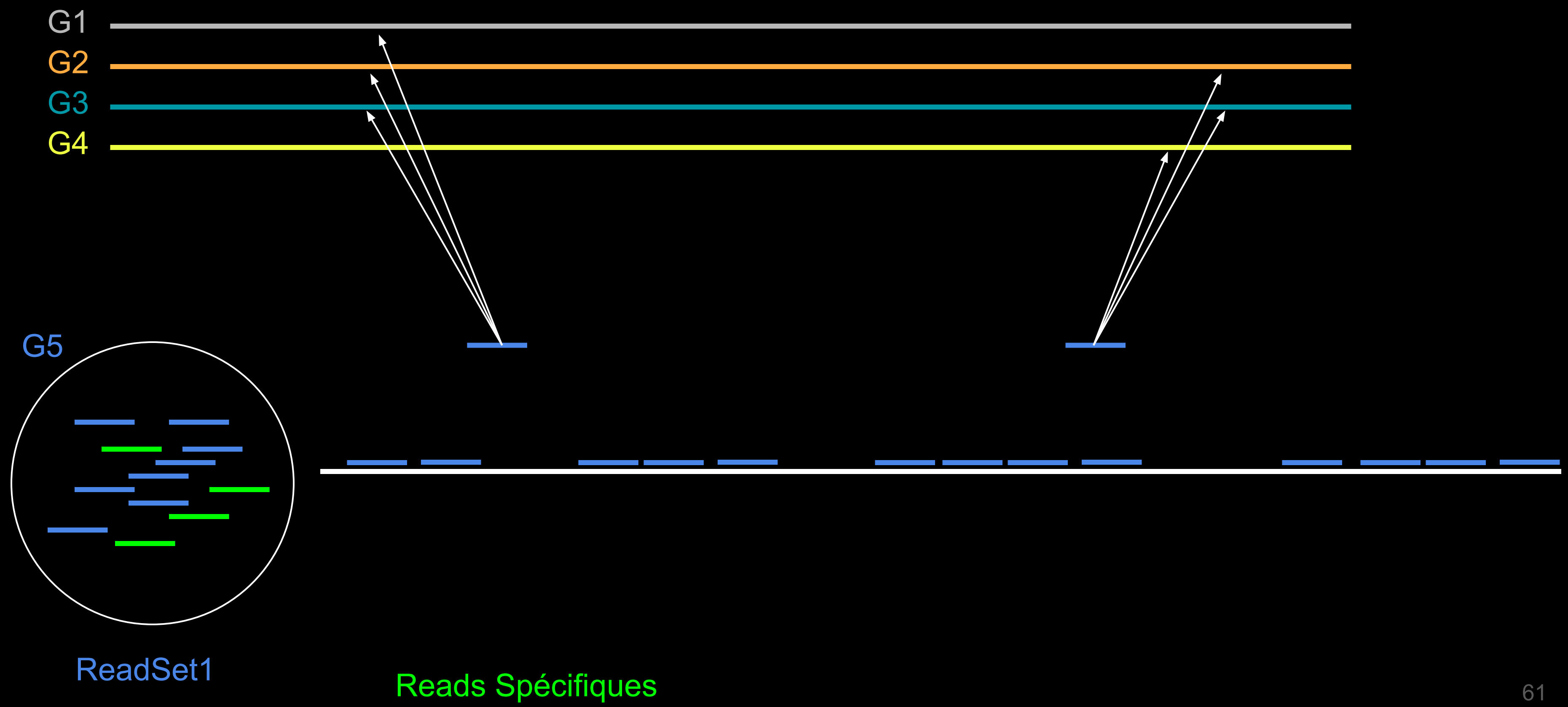

G1
G2
G3
G4
G5
ReadSet1

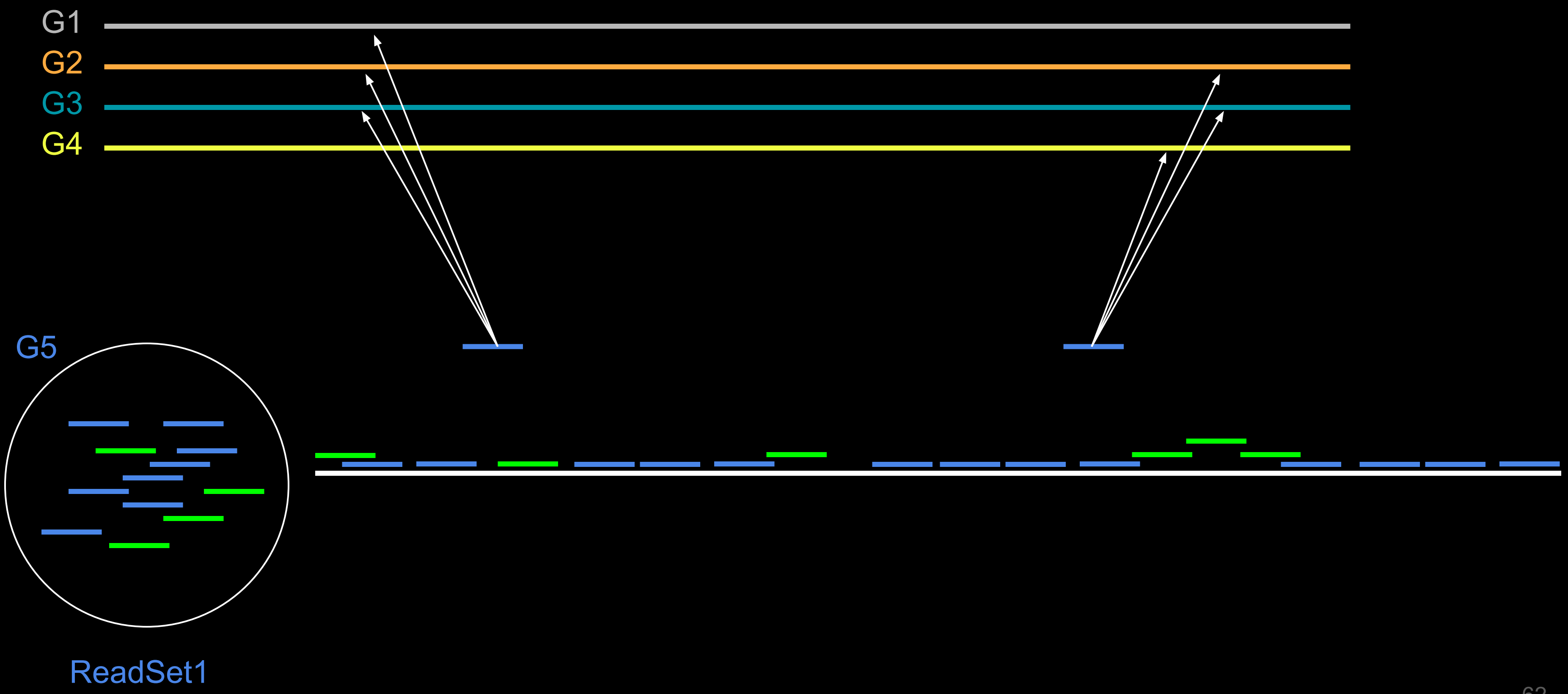

G1
G2
G3
G4
G5
ReadSet1


G1
G2
G3
G4
G5
ReadSet1

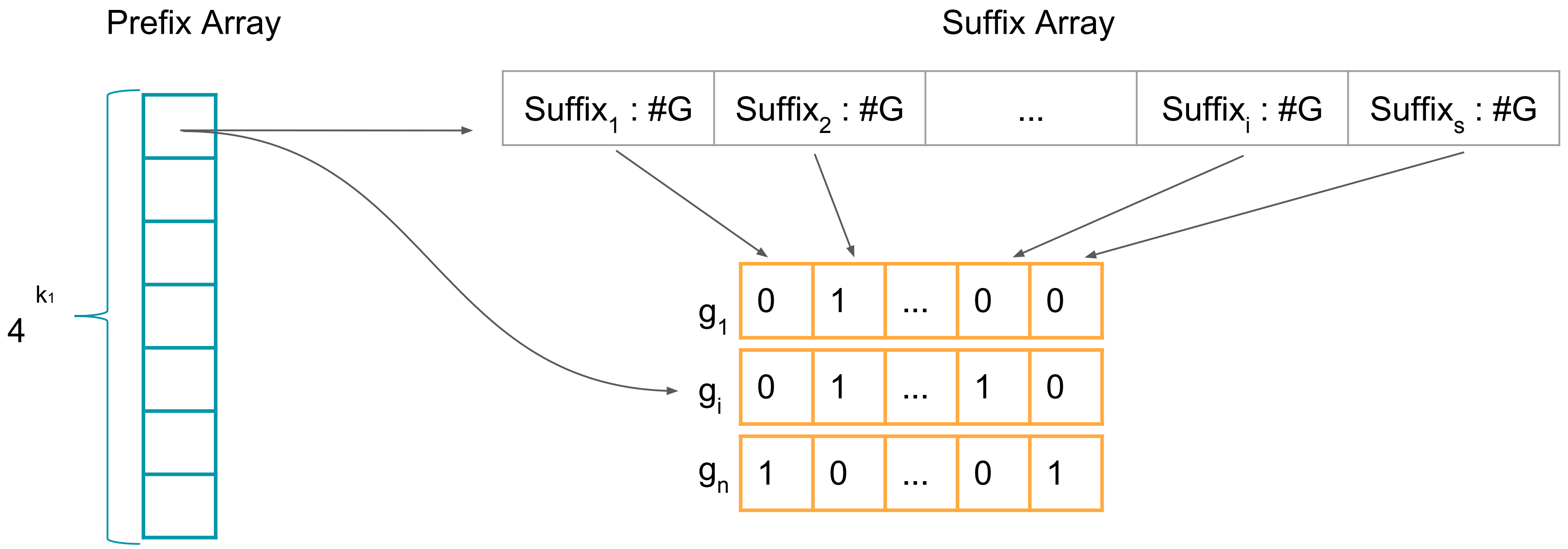


G1
G2
G3
G4
G5
ReadSet1


G1
G2
G3
G4
G5
ReadSet1
Reads Spécifiques


G1
G2
G3
G4
G5
ReadSet1

#endif



Prefix Array
Suffix Array
Suffix$_1$: #G
Suffix$_2$: #G
...
Suffix$_i$: #G
Suffix$_s$: #G
4^{k_1}
g_1
0 1 ... 0 0
g_i
0 1 ... 1 0
g_n
1 0 ... 0 1