Brown Bag Lunch #1

Introduction à Kafka-Streams

Sommaire

Sommaire

- Kafka-streams, c’est quoi ?

Sommaire

- Kafka-streams, c’est quoi ?
- Exemple #1 : to-uppercase

Sommaire

- Kafka-streams, c’est quoi ?
- Exemple #1 : to-uppercase
- Exemple #2 : word-split

Sommaire

- Kafka-streams, c’est quoi ?
- Exemple #1 : to-uppercase
- Exemple #2 : word-split
- Exemple #3 : wordcount

Sommaire

- Kafka-streams, c’est quoi ?
- Exemple #1 : to-uppercase
- Exemple #2 : word-split
- Exemple #3 : wordcount
- Bonus #1: windowing

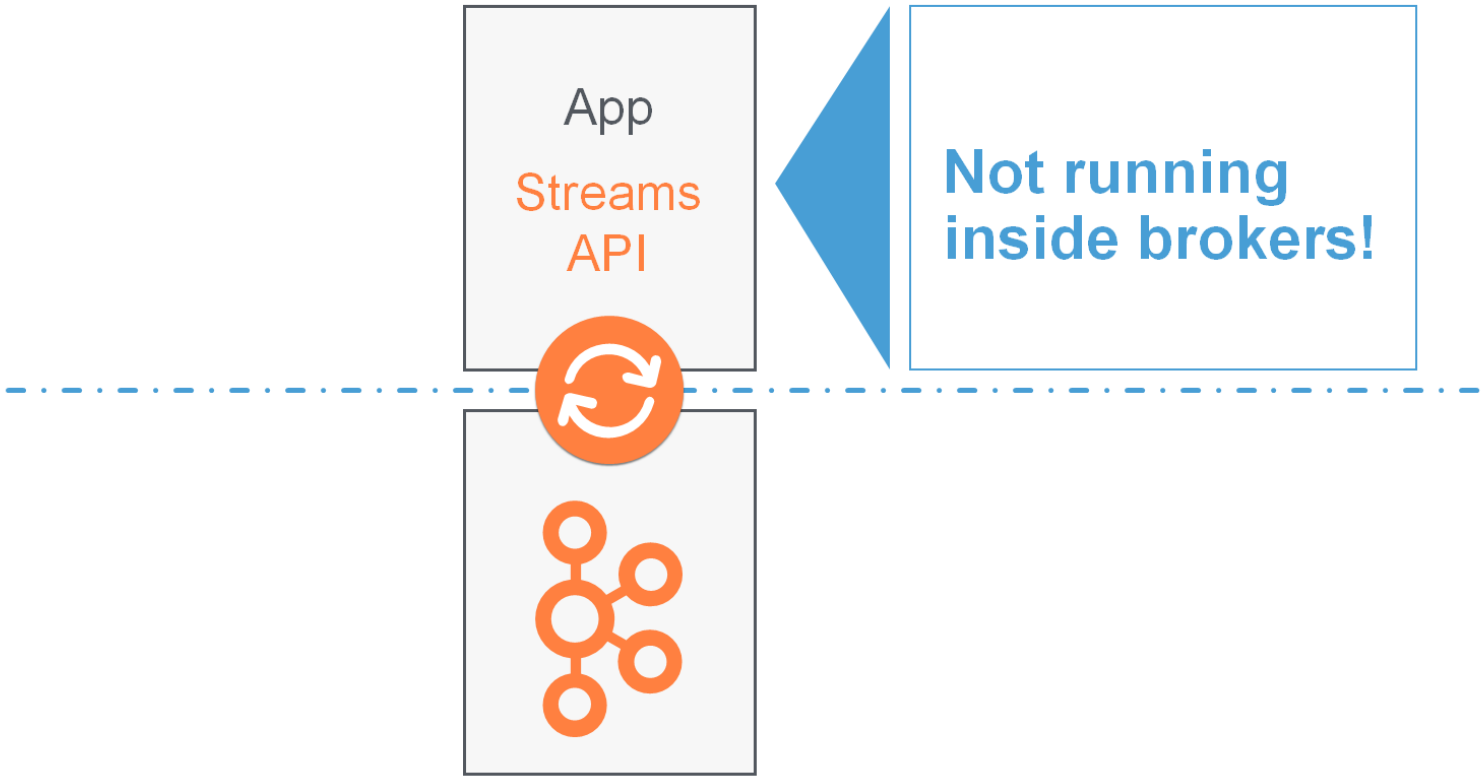
Sommaire

- Kafka-streams, c’est quoi ?
- Exemple #1 : to-uppercase
- Exemple #2 : word-split
- Exemple #3 : wordcount
- Bonus #1: windowing
- Bonus #2: joins – streams & Ktable

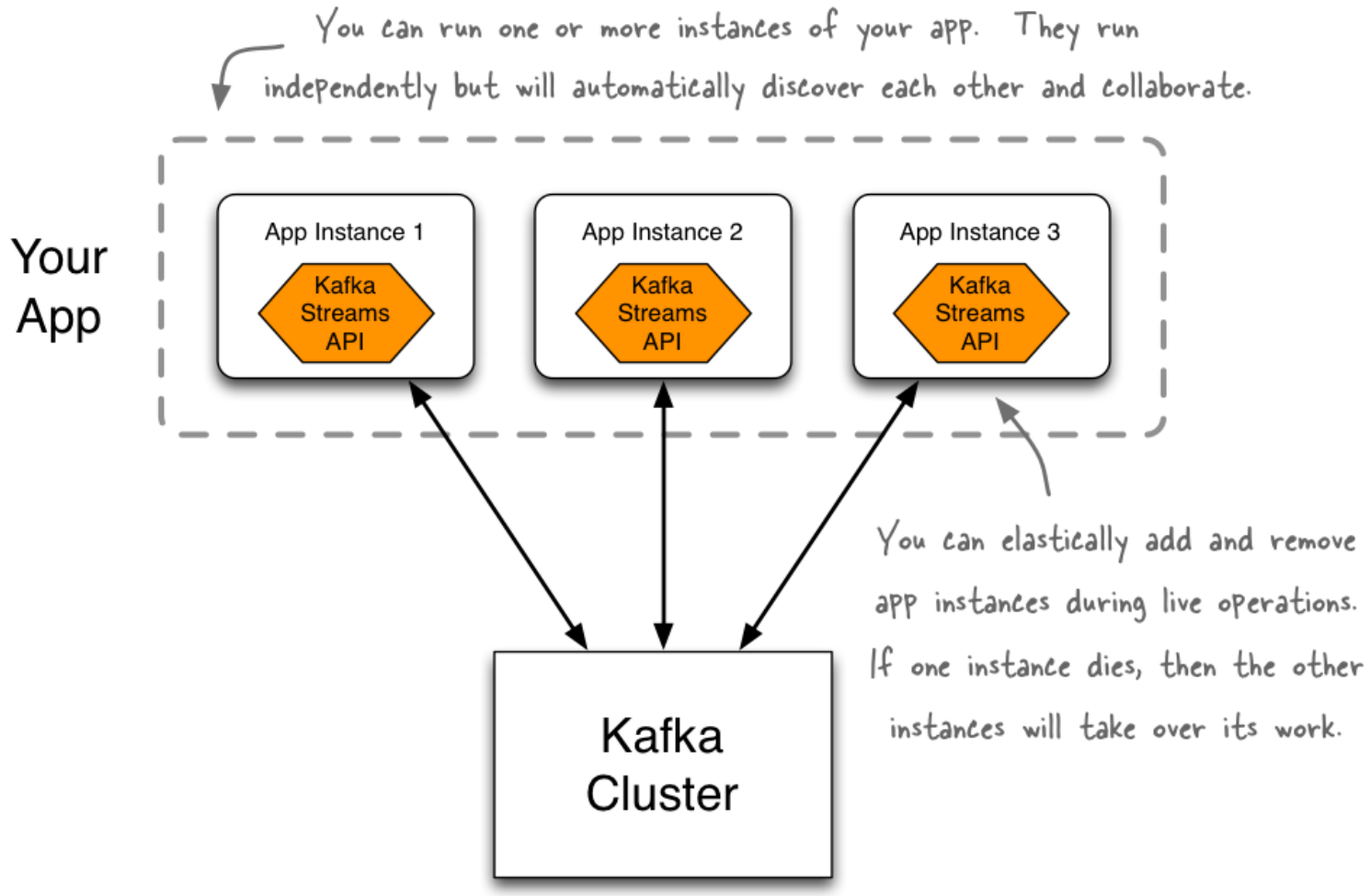
Sommaire

- Kafka-streams, c’est quoi ?
- Exemple #1 : to-uppercase
- Exemple #2 : word-split
- Exemple #3 : wordcount
- Bonus #1: windowing
- Bonus #2: joins – streams & Ktable
- Conclusions ?

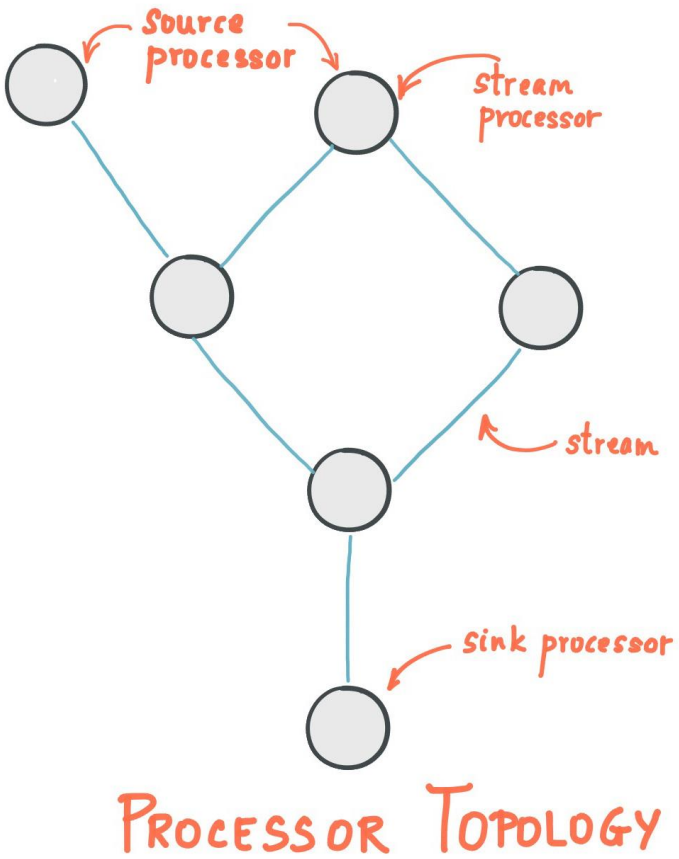
Kafka-streams, c’est quoi ? – Vue d’avion



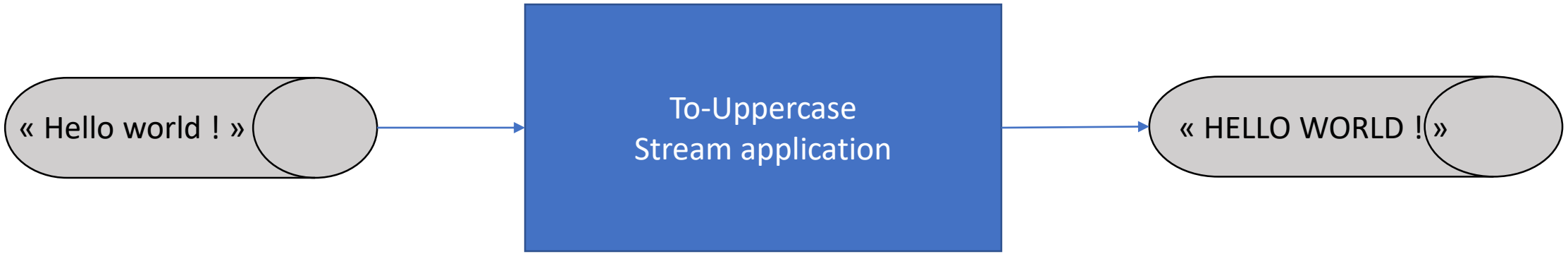
Kafka-streams, c'est quoi ? – Vue d'avion



Kafka-streams, c’est quoi ? – quelques concepts



Exemple #1 : to-uppercase



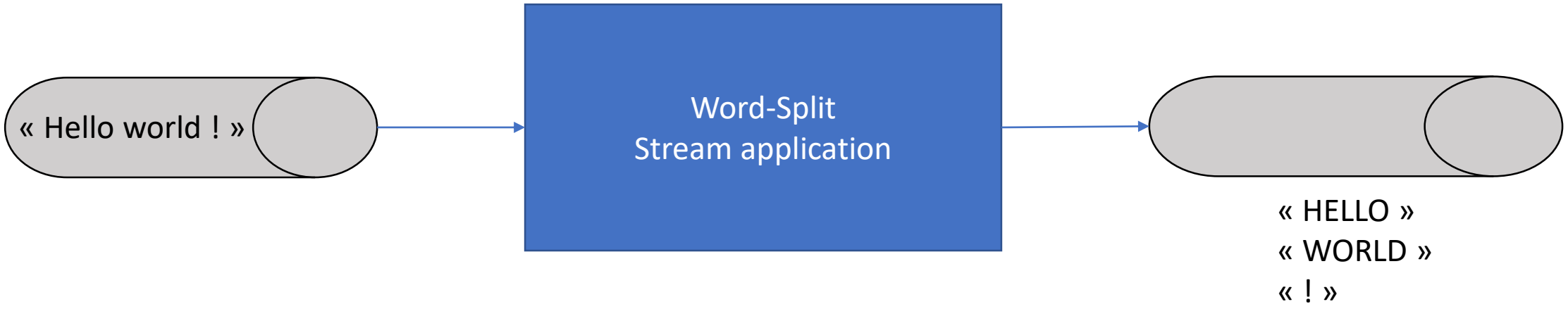
Exemple #1 : to-uppercase

- Codons !

Exemple #1 : to-uppercase

- Codons !
- Checklist
  - Créer les topics manuellement avant
  - Lister et décrire les consumer group une fois l'appli lancée et arrêtée
  - Lister les topics
  - Tests unitaires
  - Implémentation via la DSL fonctionnelle de « haut niveau »
  - Implémentation via l'API impérative de « bas niveau »
  - Vue « Tasks et partitions » de l'exemple
    - Augmenter le nombre de partitions et constater l'augmentation du nombre de tasks
  - Step-by-step de l'exemple « architecture » de la doc de Confluent

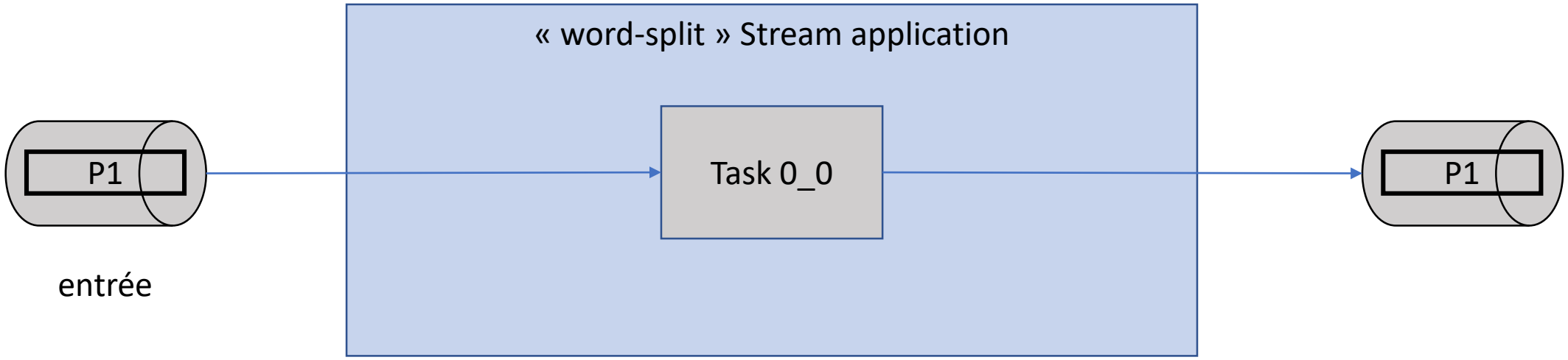
Exemple #2 : word-split



Exemple #2 : word-split

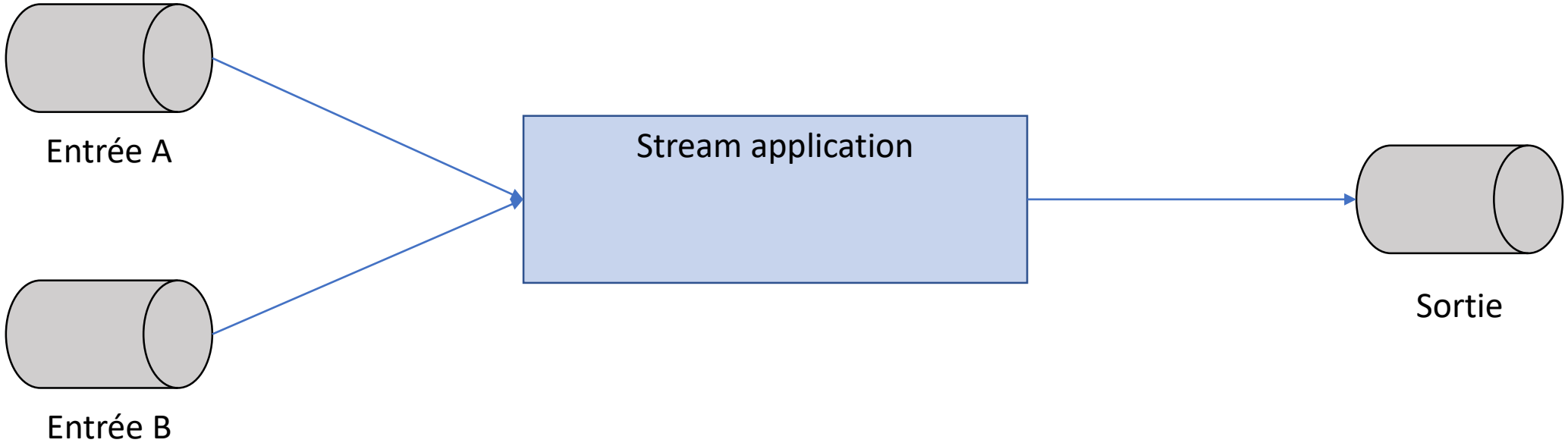
- Codons !

Exemple #2 : word-split

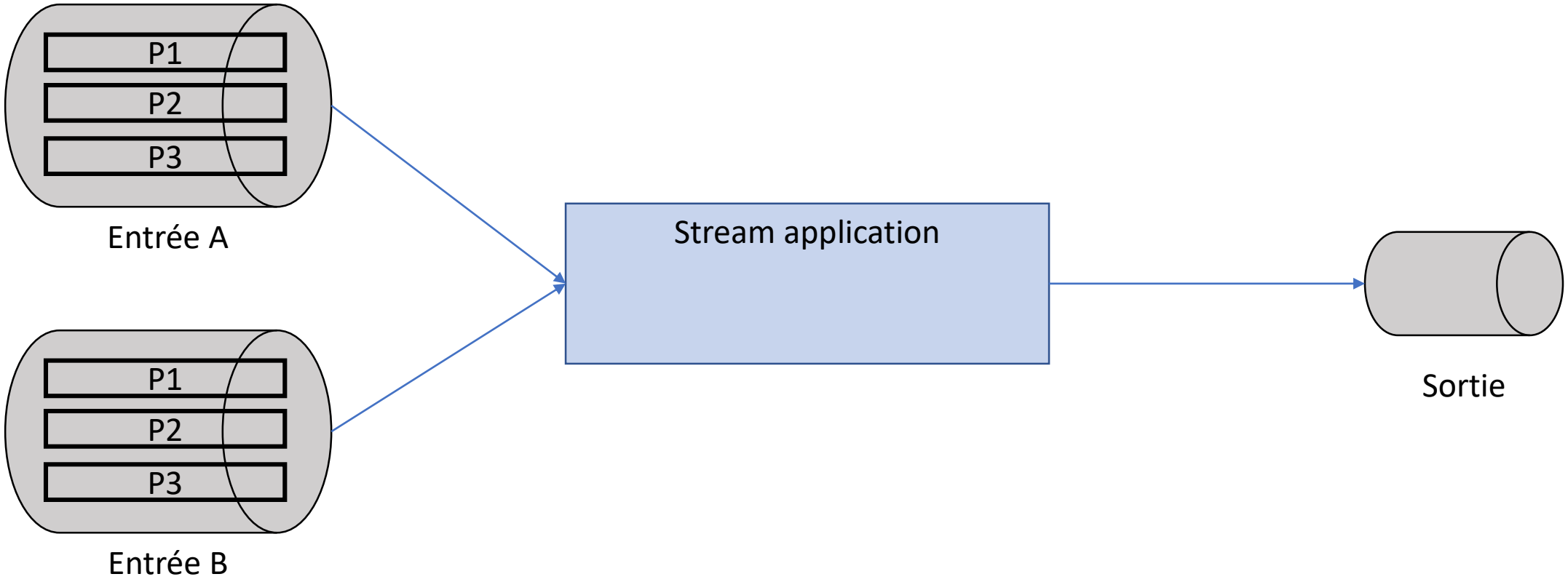


Interlude: les « tasks » et leur affectation

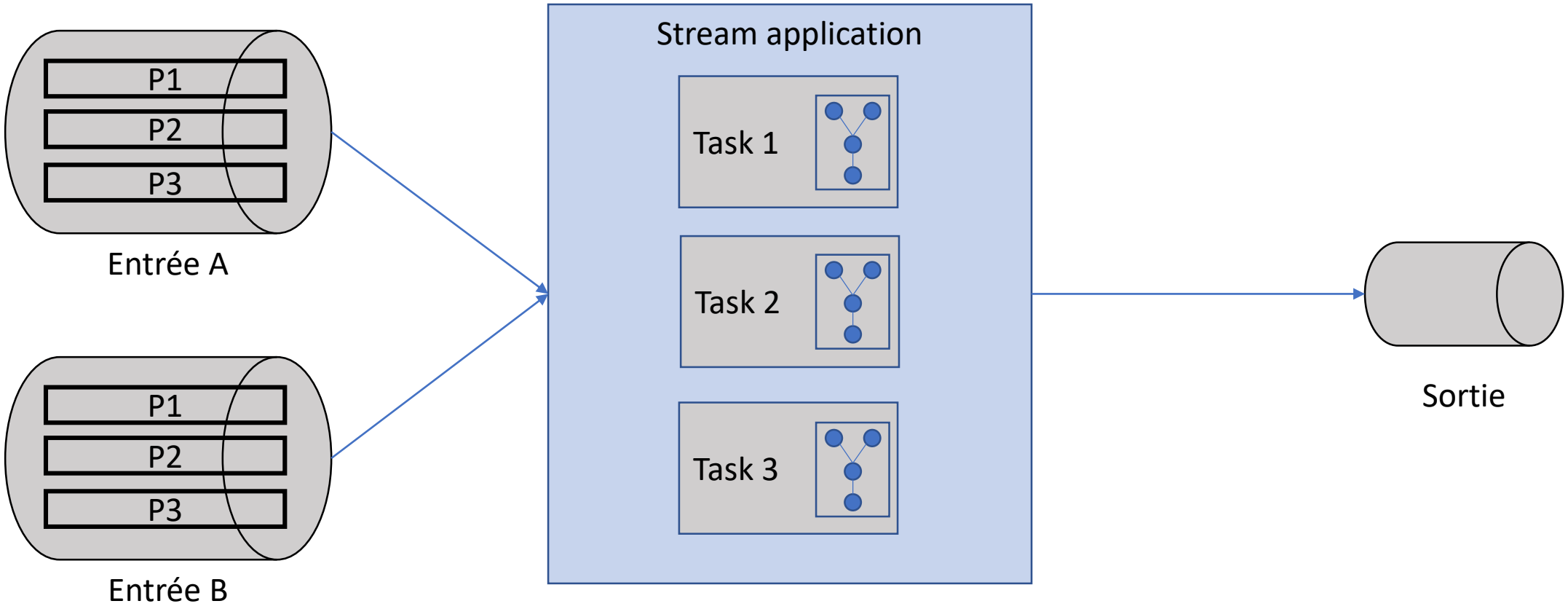
Interlude: les « tasks » et leur affectation



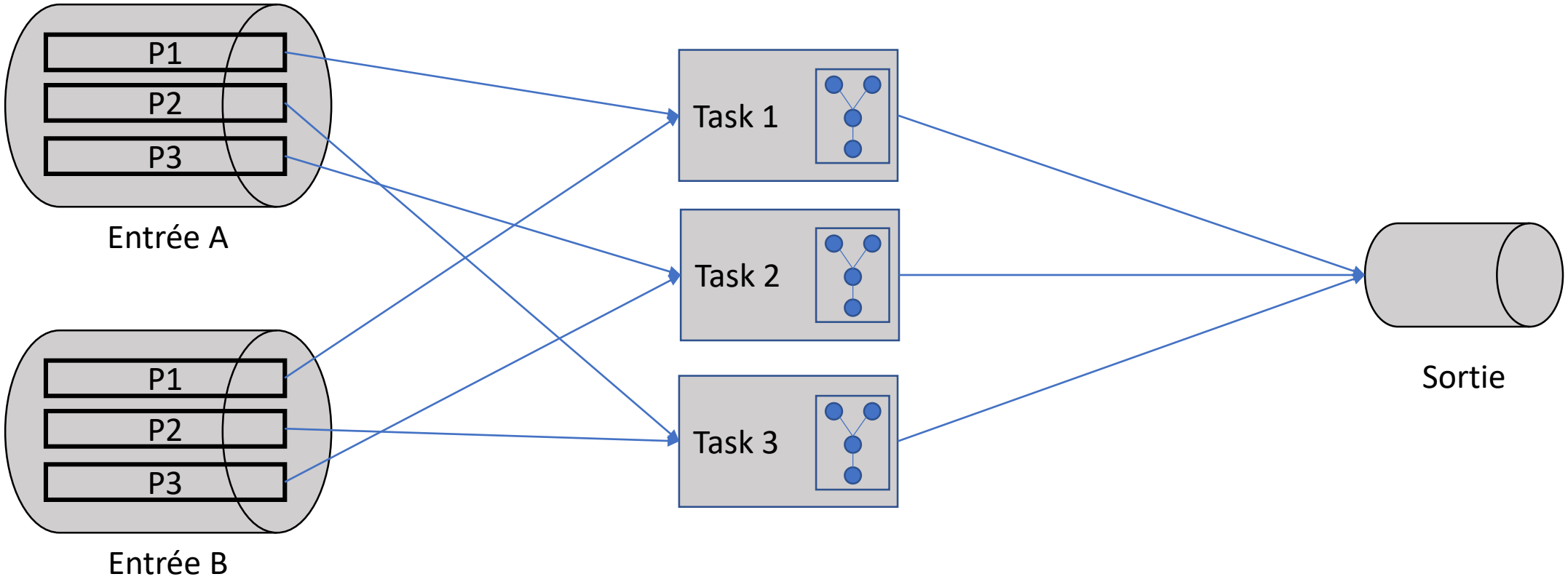
Interlude: les « tasks » et leur affectation



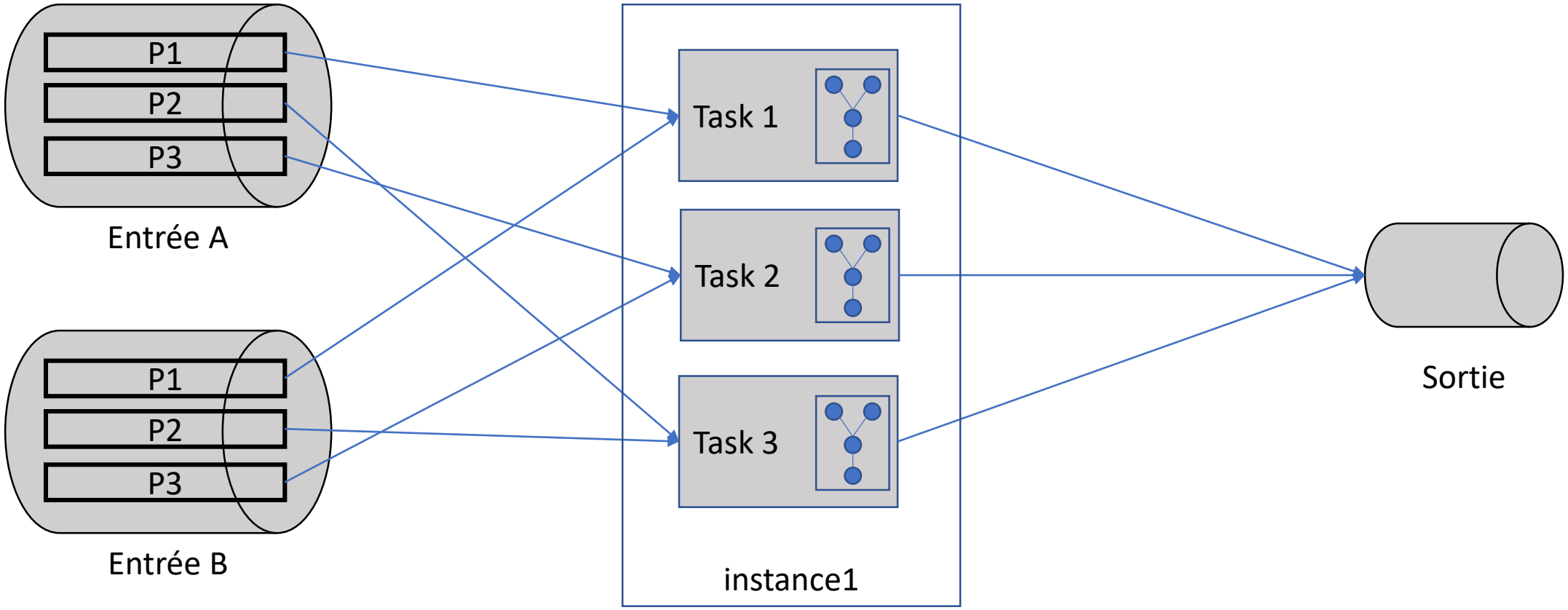
Interlude: les « tasks » et leur affectation



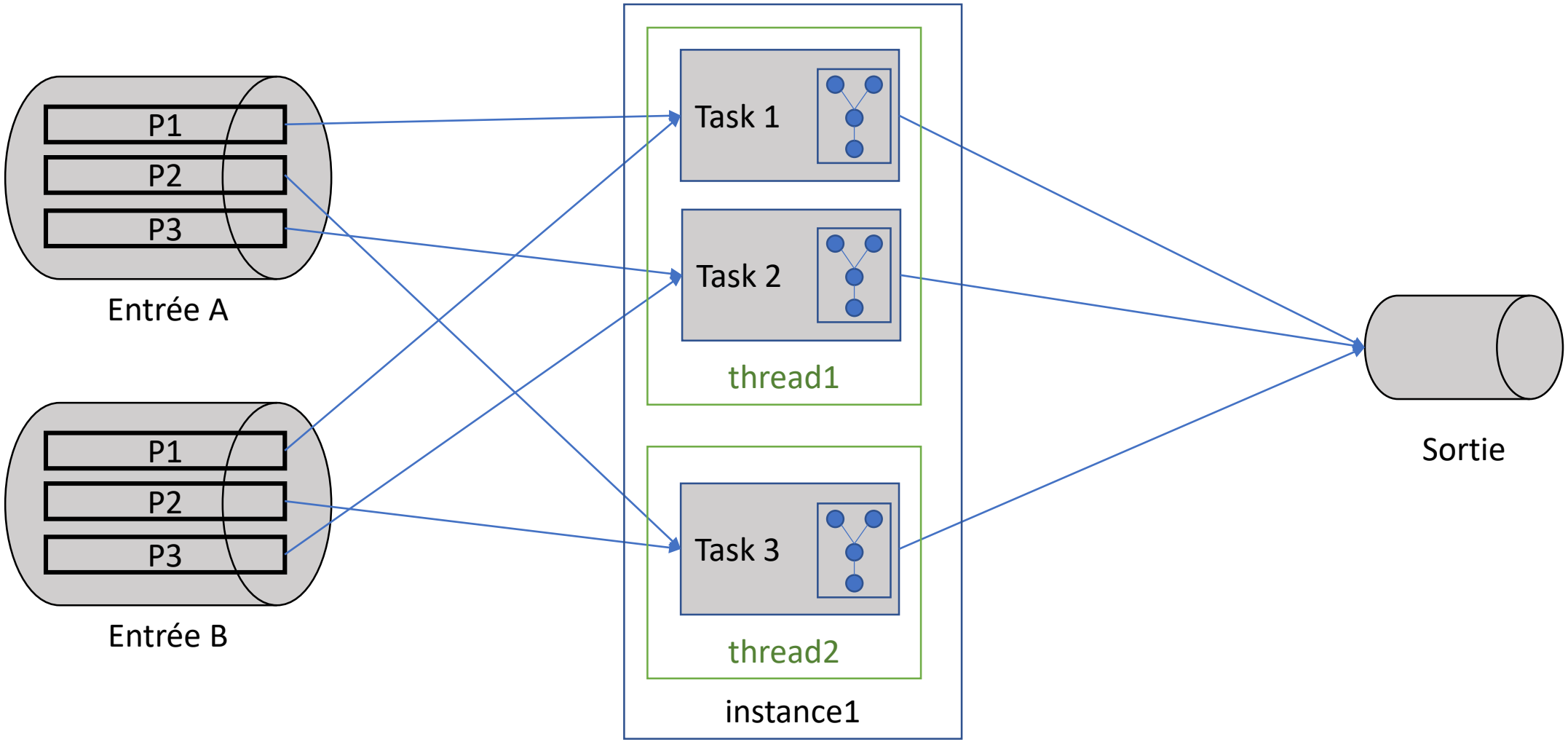
Interlude: les « tasks » et leur affectation



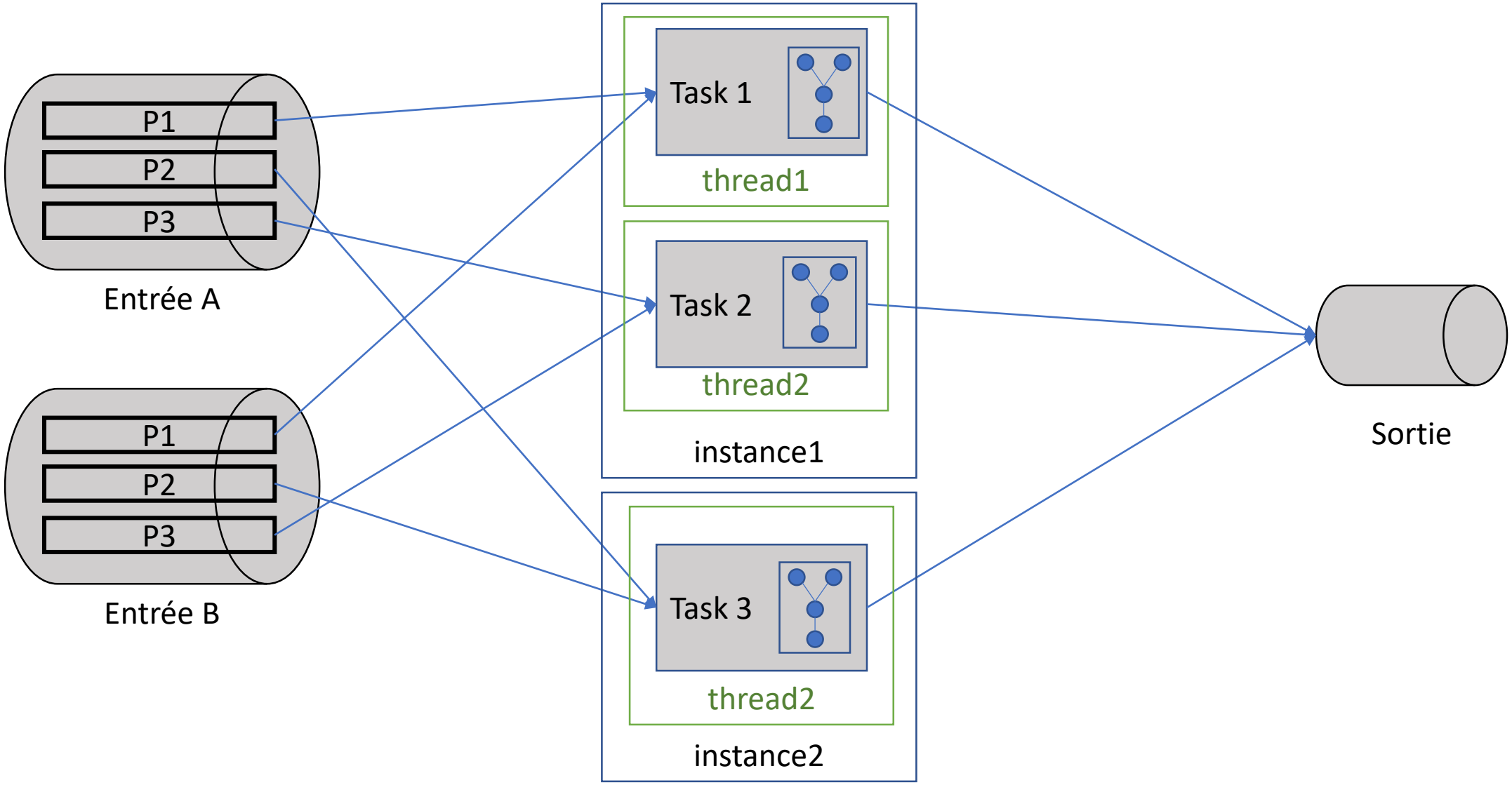
Interlude: les « tasks » et leur affectation



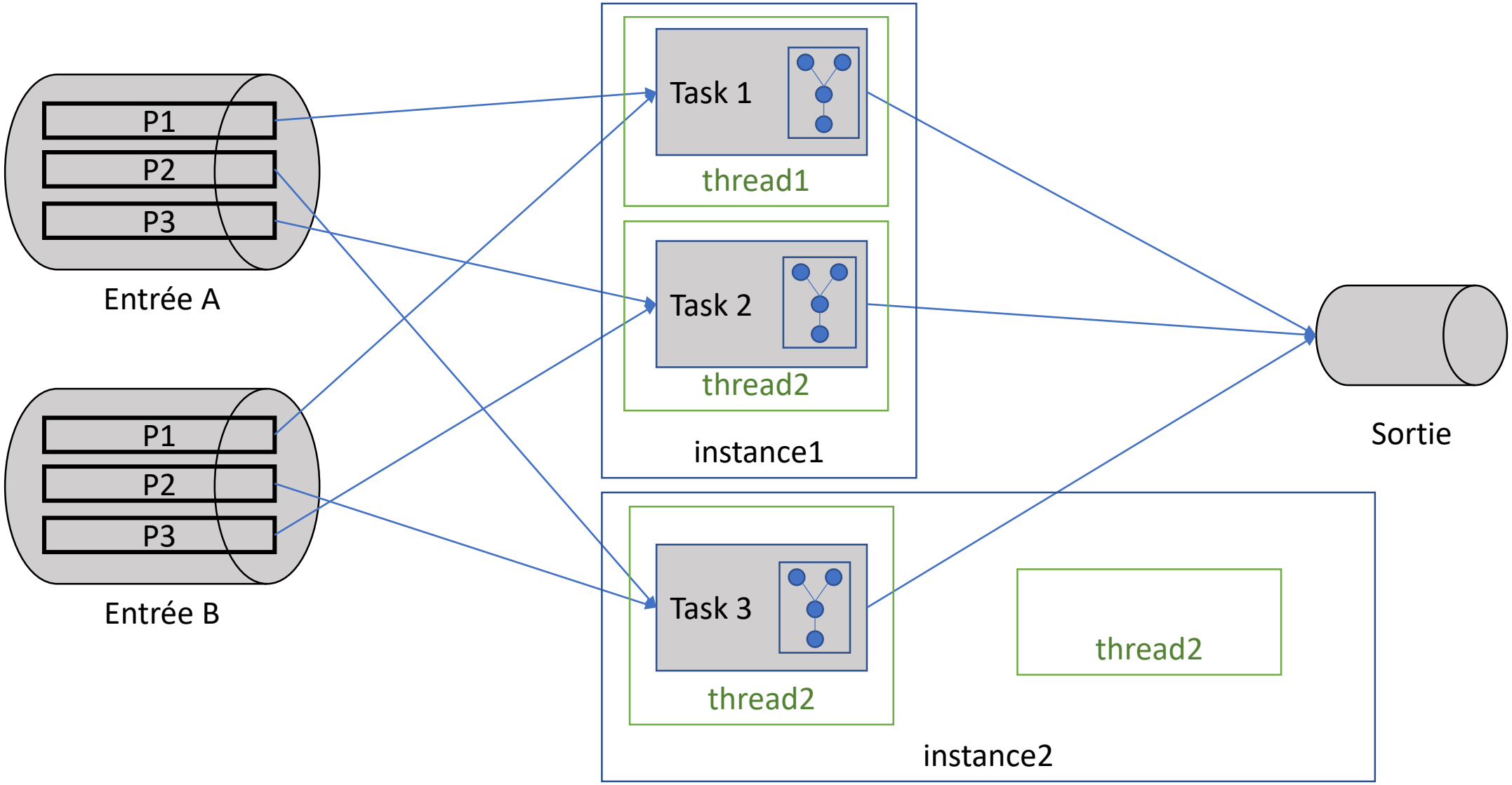
Interlude: les « tasks » et leur affectation



Interlude: les « tasks » et leur affectation



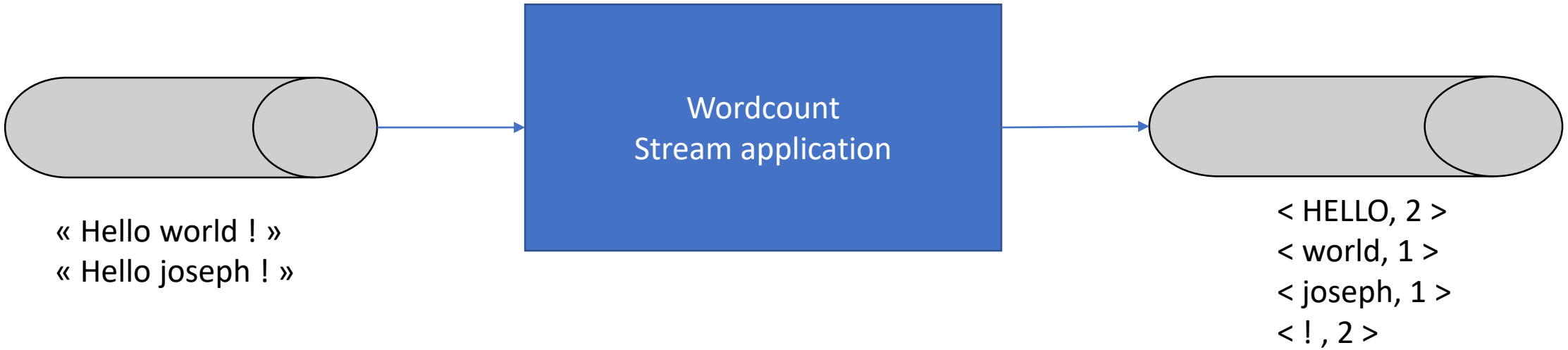
Interlude: les « tasks » et leur affectation



Exemple #2 : word-split

- Codons !
- Checklist
  - Créer les topics manuellement avant
  - Tests unitaires
  - Implémentation via la DSL fonctionnelle de « haut niveau »
  - Implémentation via l'API impérative de « bas niveau »
  - Vue « tasks et partition » très rapidement

Exemple #3 : wordcount



Exemple #3 : wordcount – illustration de l’algo

Exemple #3 : wordcount – illustration de l’algo

< Ø, première ligne >
< Ø, deuxième ligne >
< Ø, troisième ligne >

Exemple #3 : wordcount – illustration de l'algo

< Ø, première ligne >
< Ø, deuxième ligne >
< Ø, troisième ligne >

flatmap

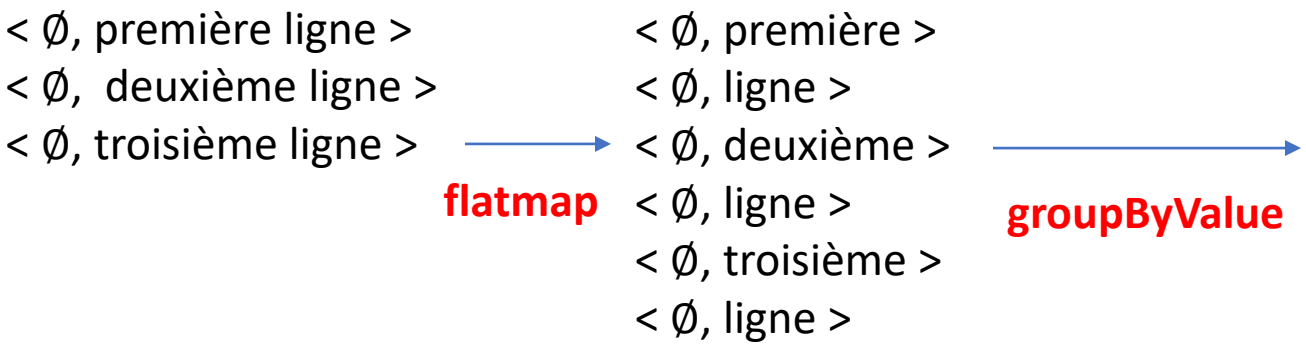
Exemple #3 : wordcount – illustration de l'algo

< Ø, première ligne >
< Ø, deuxième ligne >
< Ø, troisième ligne >

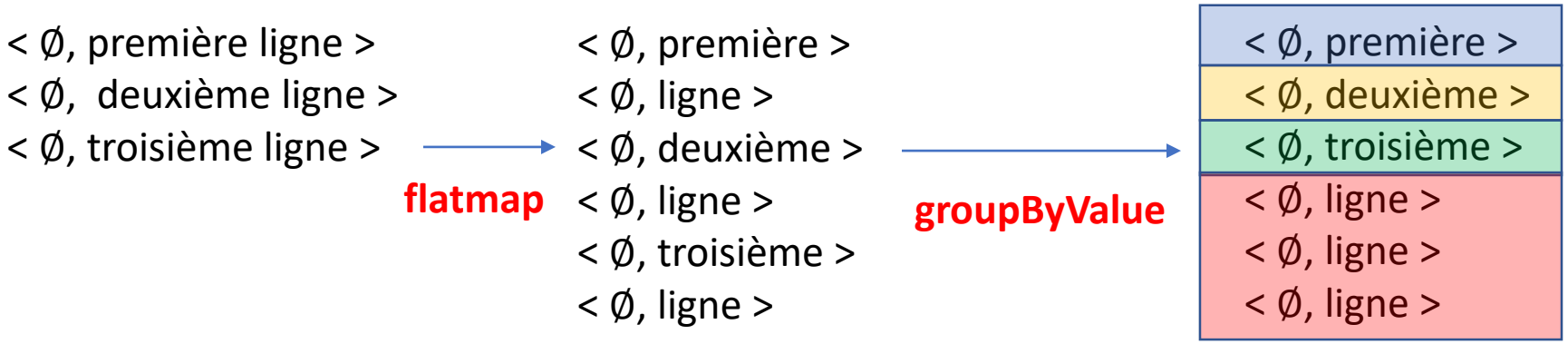
flatmap

< Ø, première >
< Ø, ligne >
< Ø, deuxième >
< Ø, ligne >
< Ø, troisième >
< Ø, ligne >

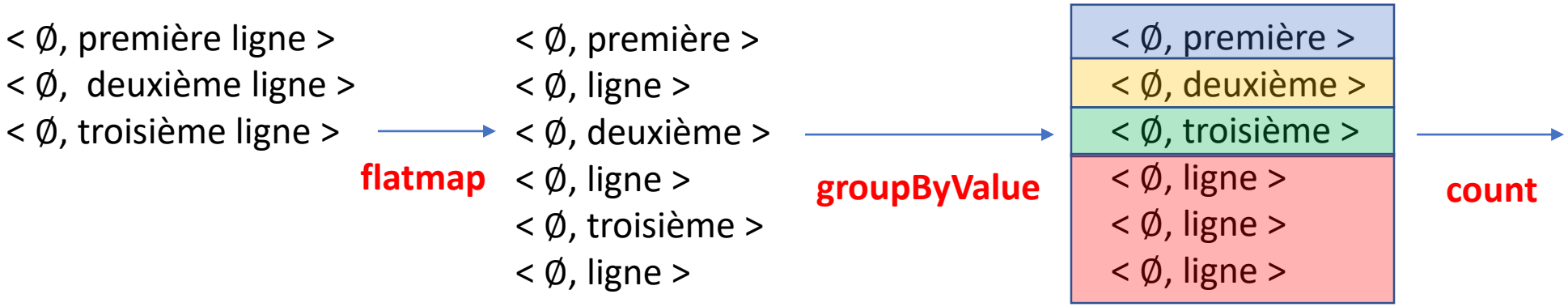
Exemple #3 : wordcount – illustration de l'algo



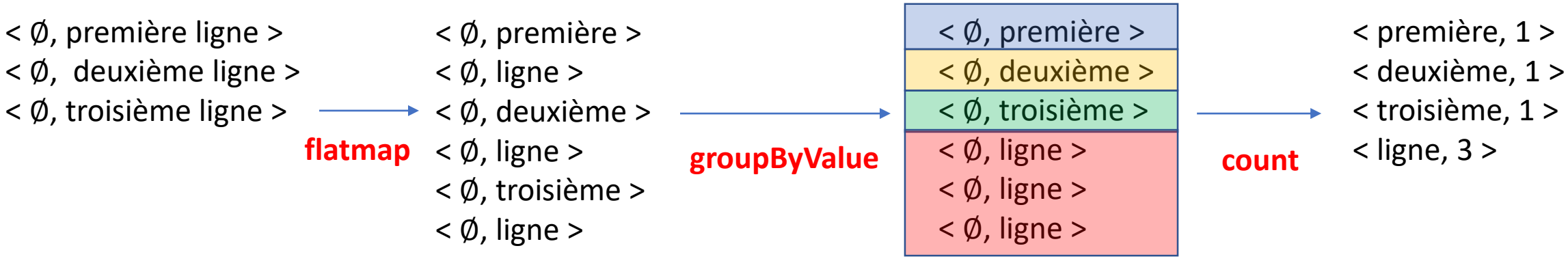
Exemple #3 : wordcount – illustration de l'algo



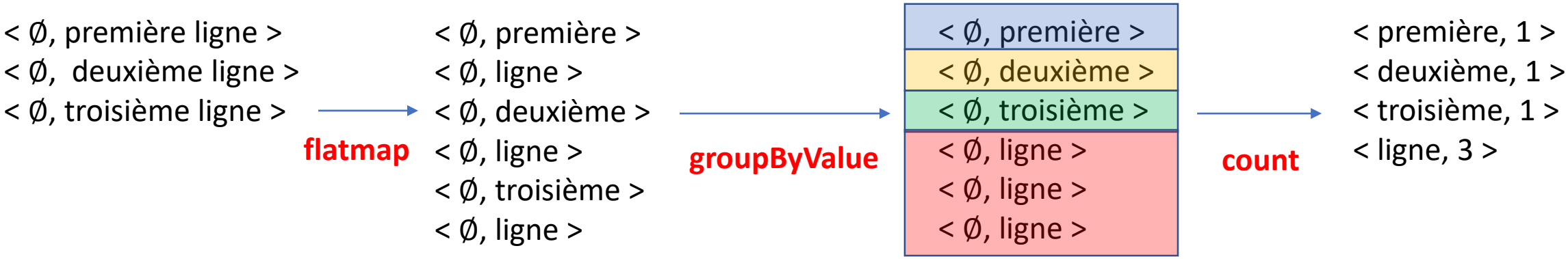
Exemple #3 : wordcount – illustration de l'algo



Exemple #3 : wordcount – illustration de l'algo



Exemple #3 : wordcount – illustration de l'algo



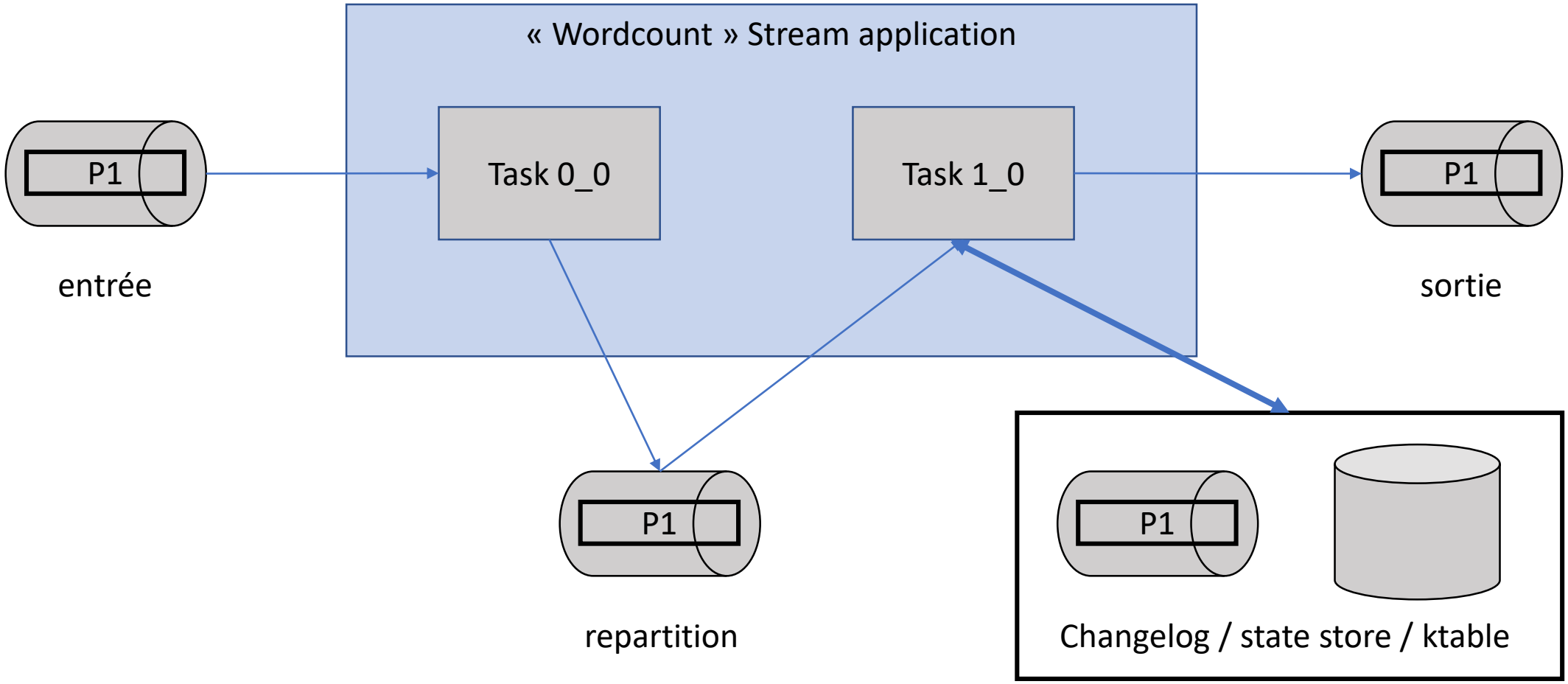
L'intuition de l'analogie avec l'algo MapReduce était en fait tout à fait correcte, l'étude du détail de l'implémentation et de la variante « group by key » le révèlera encore plus.

Exemple #3 : wordcount – Codons !

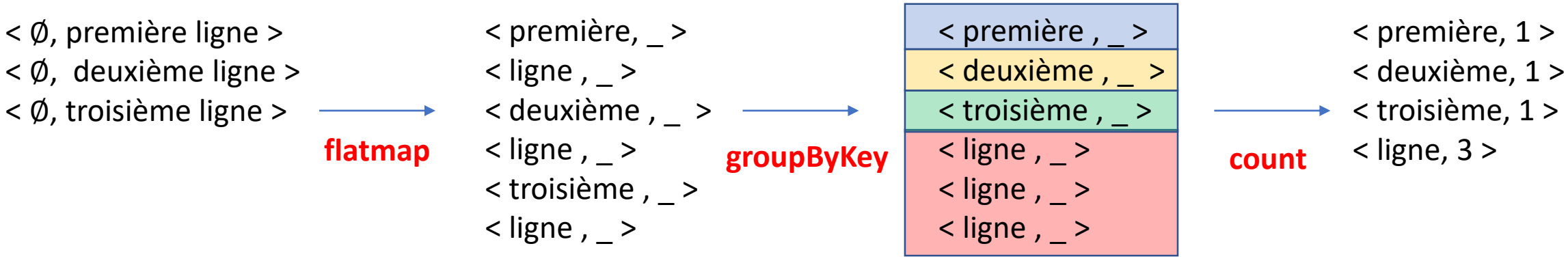
Exemple #3 : wordcount – Codons !

- Checklist
  - Créer les topics manuellement avant
  - Tests unitaires ?
  - groupByValue
  - Lister les topics créés, focus sur le topic « changelog »
    - State store, dualité Ktable / Kstream, Fault Tolerance
    - « repartition » vide pour le moment, on le verra dans l'exemple suivant

Exemple #3 : wordcount

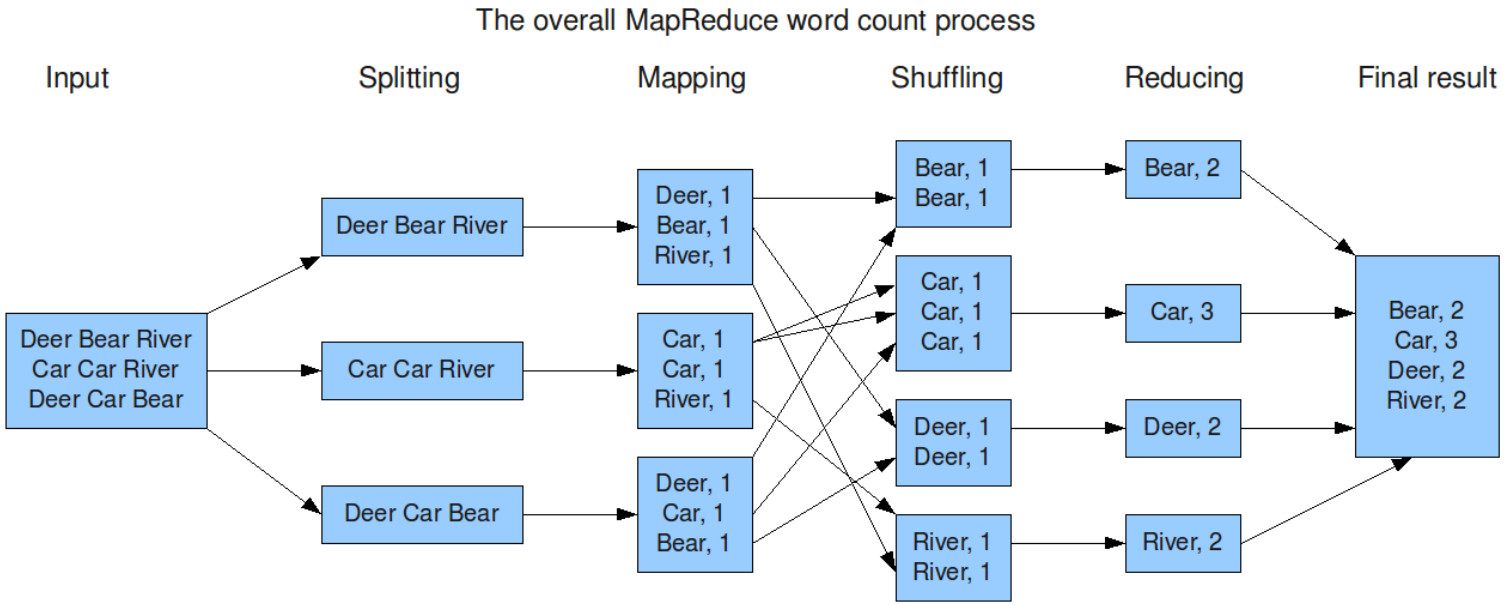


Exemple #3 : wordcount – illustration de la variante « group by key »

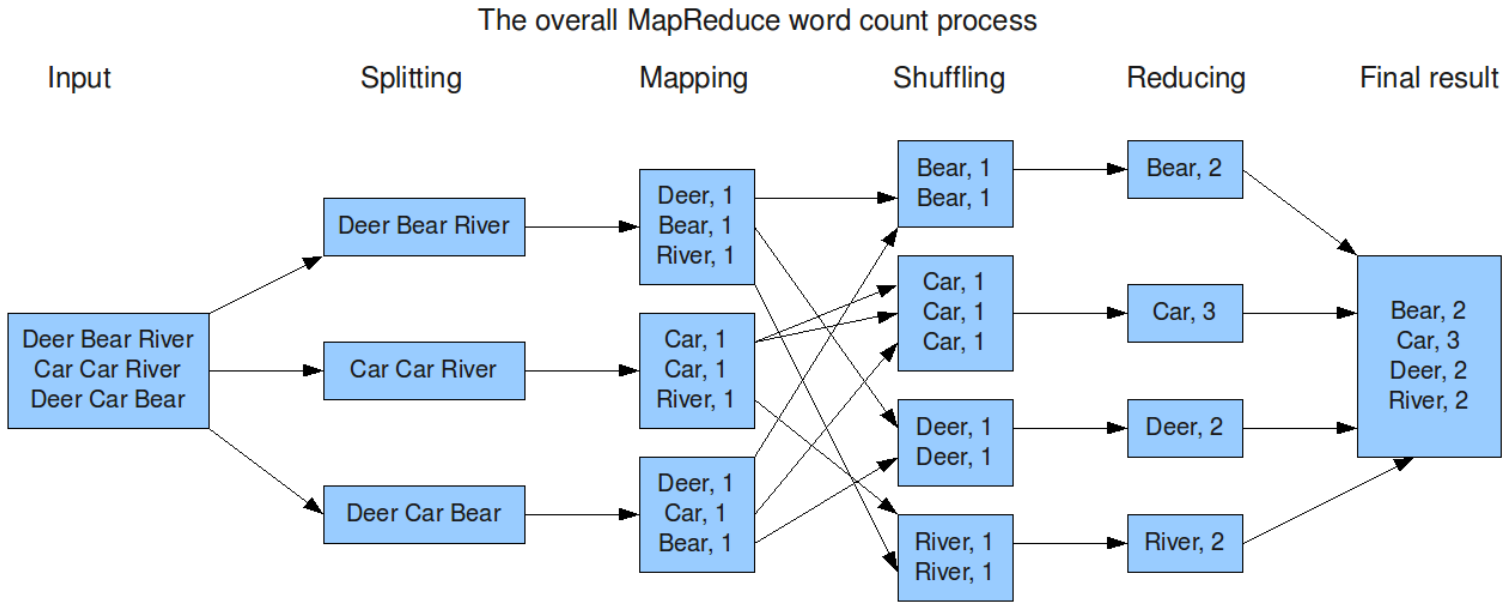


Exemple #3 : wordcount – illustration de la variante « group by key » – codons

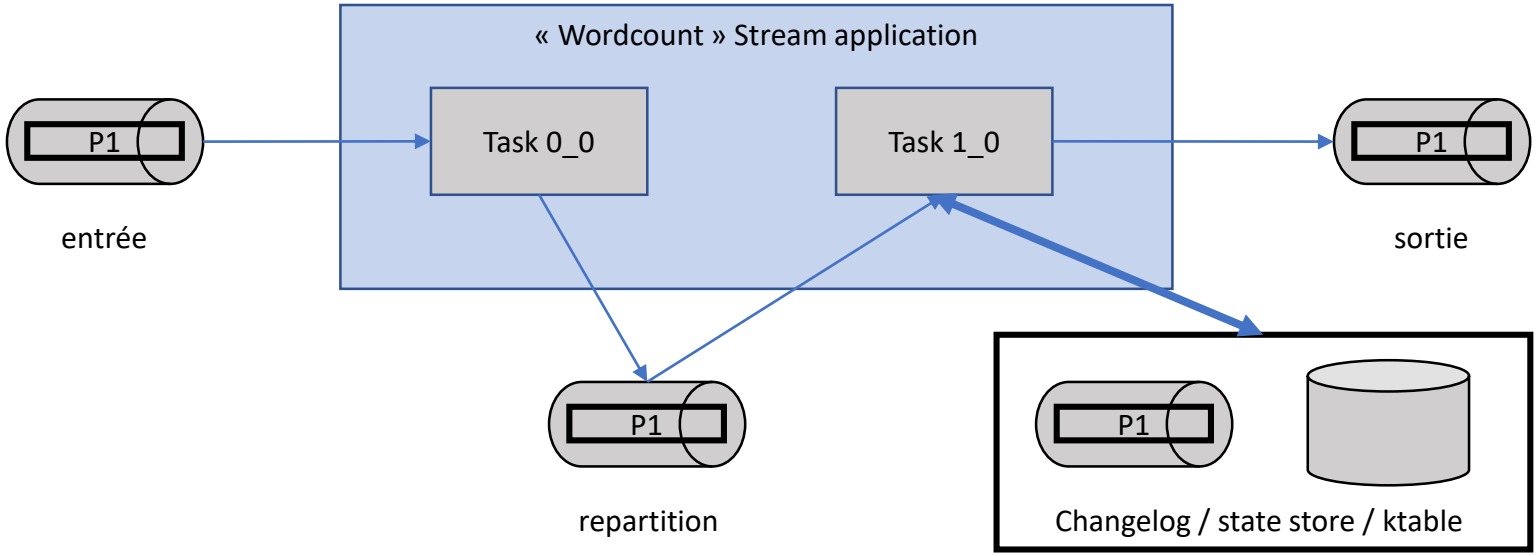
Exemple #3 : wordcount – analogie avec mapreduce



Exemple #3 : wordcount – analogie avec mapreduce




```
public static Topology getTopology() {
  StreamsBuilder builder = new StreamsBuilder();
  builder.<String, String>stream(INPUT_TOPIC)
      .flatMapValues(value -> Arrays.asList(value.split("\\W+")))
      .map((key, value) -> new KeyValue<>(value, 1))
      .groupByKey(Grouped.valueSerde(Serdes.Integer()))
      .count();
  return builder.build();
}
```



Exemple #3 : wordcount – Interactive queries

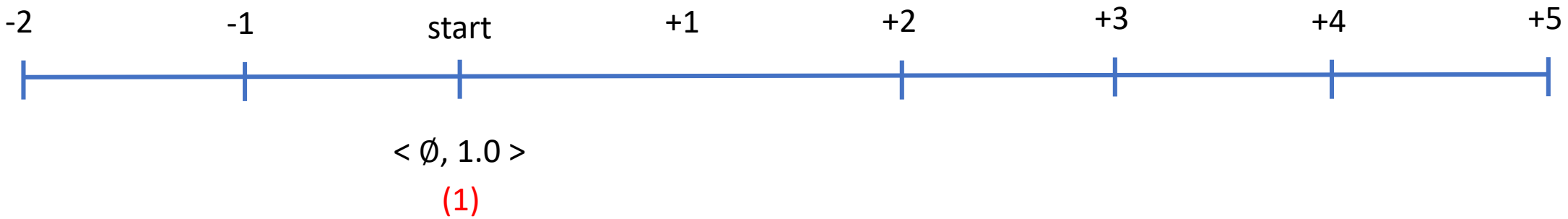
- Codons !
- Checklist
  - groupByKey
    - serdes error
    - serdes correction
  - Interactive queries
  - State stores et dualité kstream / ktable
  - Vue « tasks » des différentes implémentations

Bonus #1: windowing - Sommaire

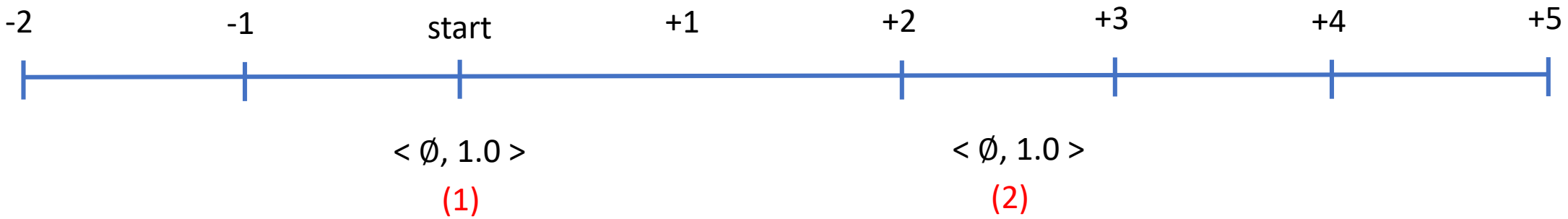
- Un premier exemple jouet
  - Sliding window
- Les différents types de windowing

Bonus #1: windowing – exemple jouet – Sliding window

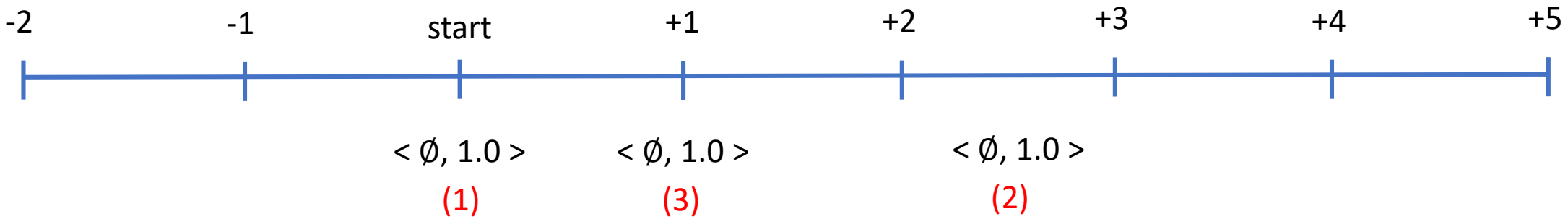
Bonus #1: windowing – exemple jouet – Sliding window



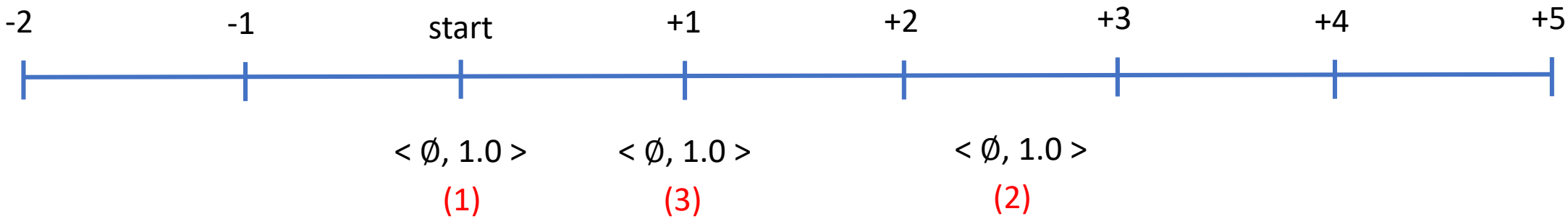
Bonus #1: windowing – exemple jouet – Sliding window



Bonus #1: windowing – exemple jouet – Sliding window

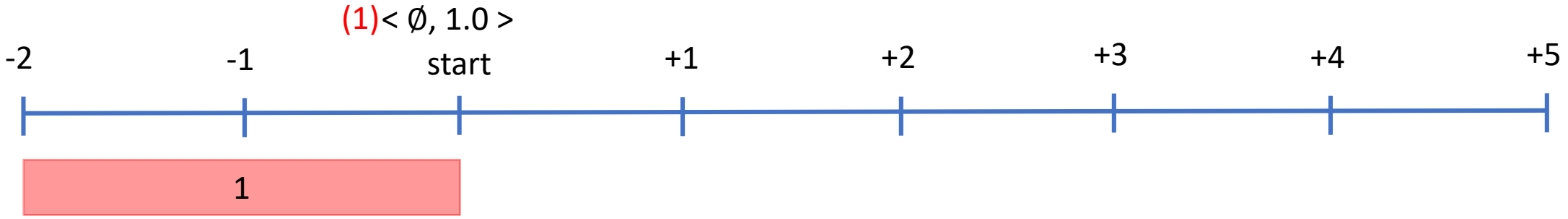


Bonus #1: windowing – exemple jouet – Sliding window

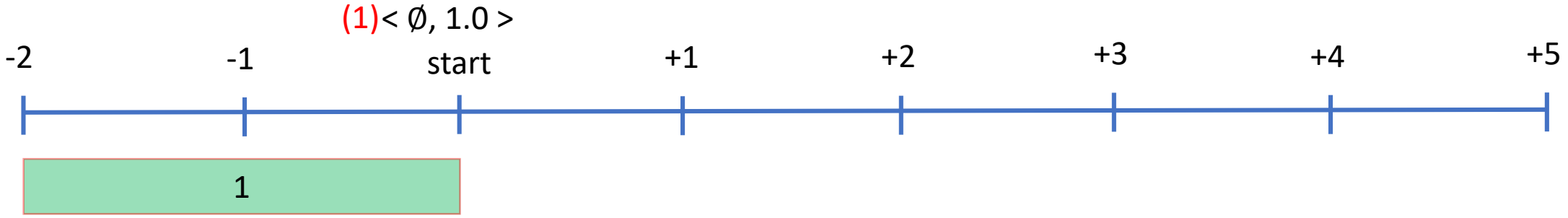



```
public static Topology getTopology() {
  StreamsBuilder builder = new StreamsBuilder();
  builder.<String, Float>stream(INPUT_TOPIC)
      .groupByKey()
      .windowedBy(SlidingWindows.withTimeDifferenceAndGrace(Duration.ofMillis(WINDOW_SIZE_MILLIS), Duration.ofMinutes(1)))
      .reduce(Float::sum, Materialized.as(STORE_NAME))
      .toStream()
      .to(OUTPUT_TOPIC, Produced.keySerde(WindowedSerdes.timeWindowedSerdeFrom(String.class, WINDOW_SIZE_MILLIS)));
  return builder.build();
}
```

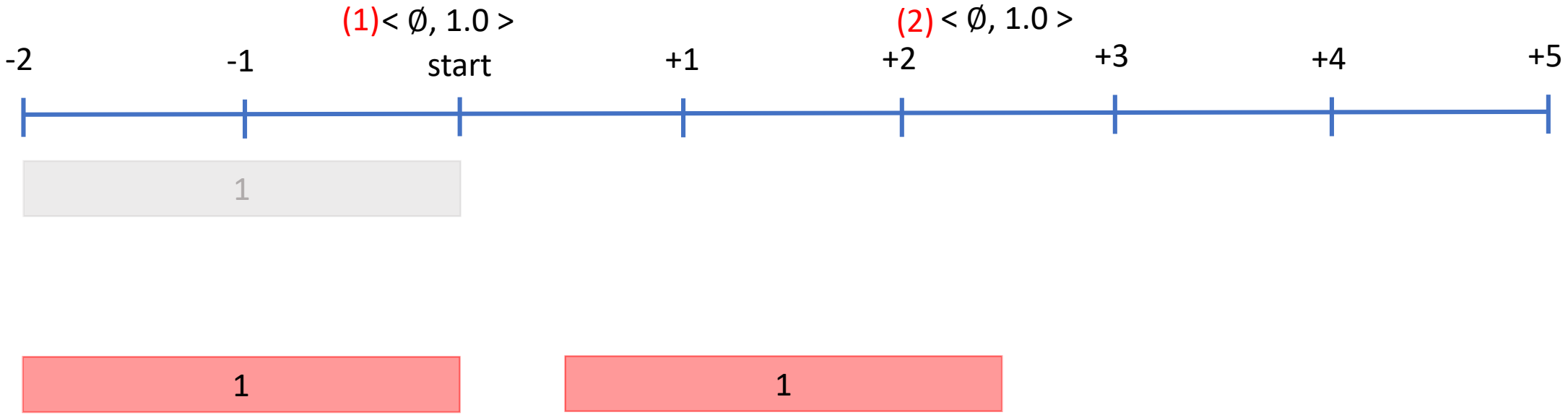
Bonus #1: windowing – exemple jouet – Sliding window



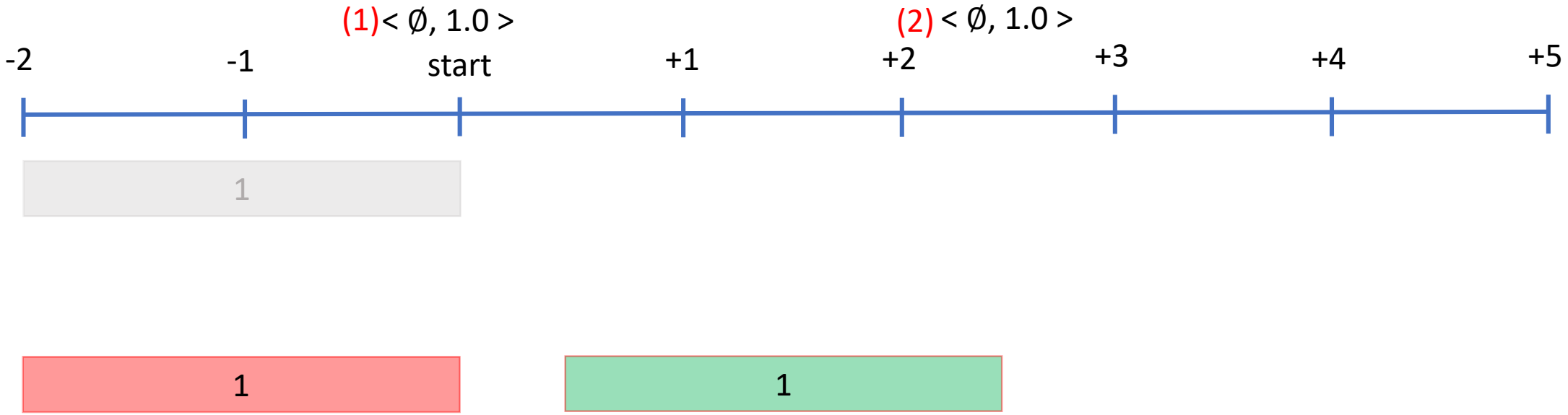
Bonus #1: windowing – exemple jouet – Sliding window



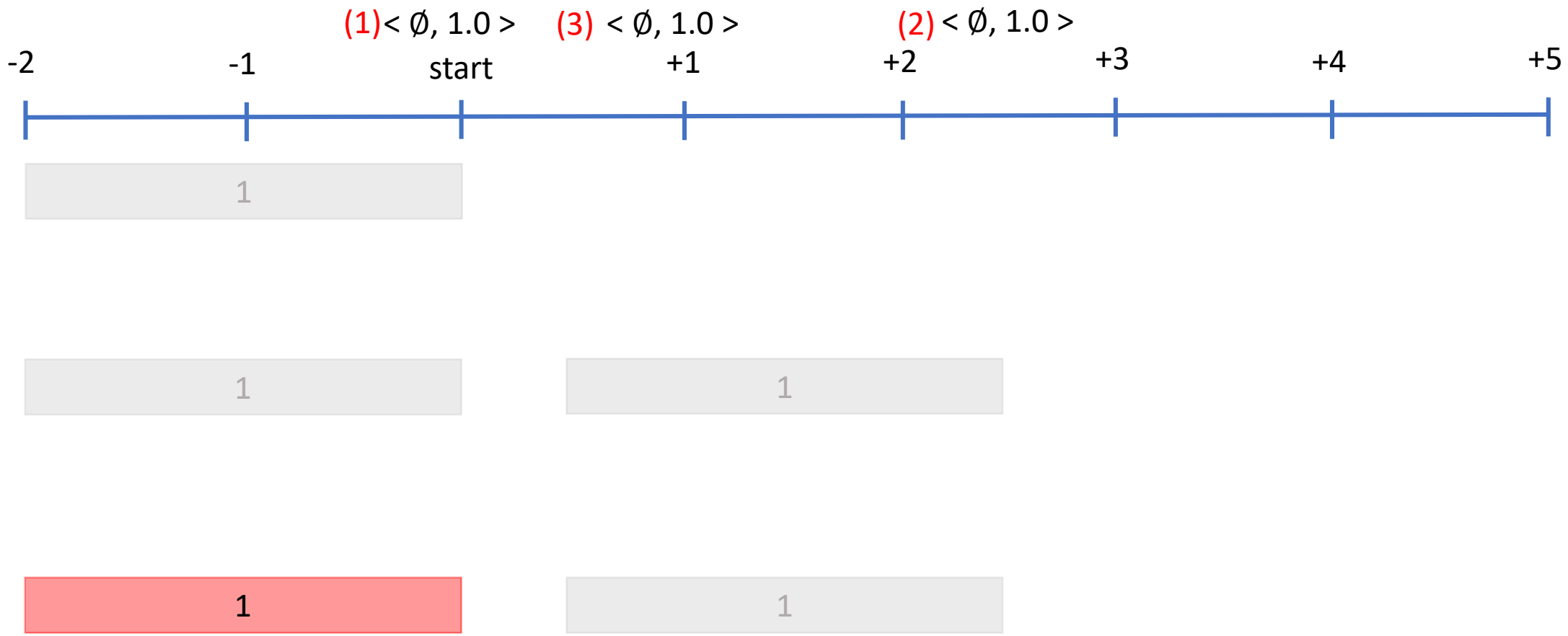
Bonus #1: windowing – exemple jouet – Sliding window



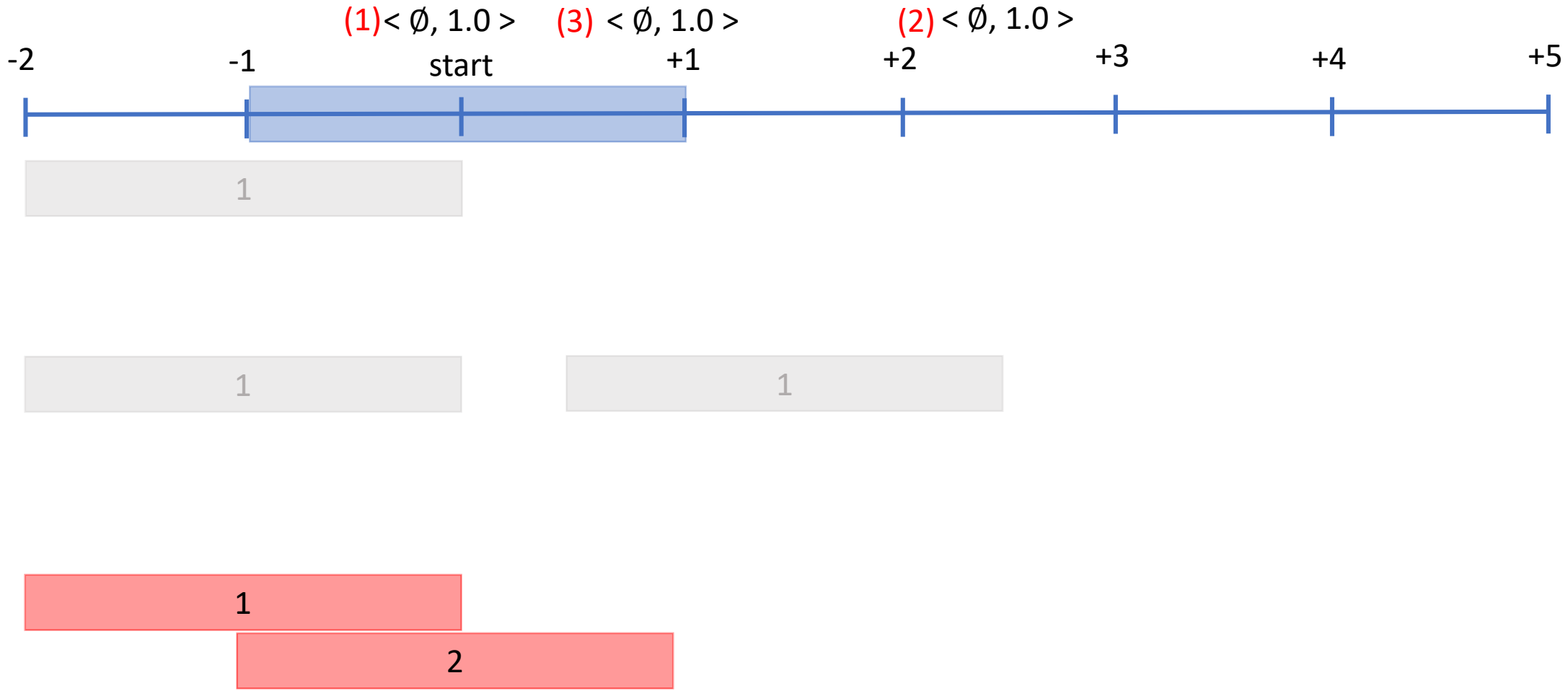
Bonus #1: windowing – exemple jouet – Sliding window



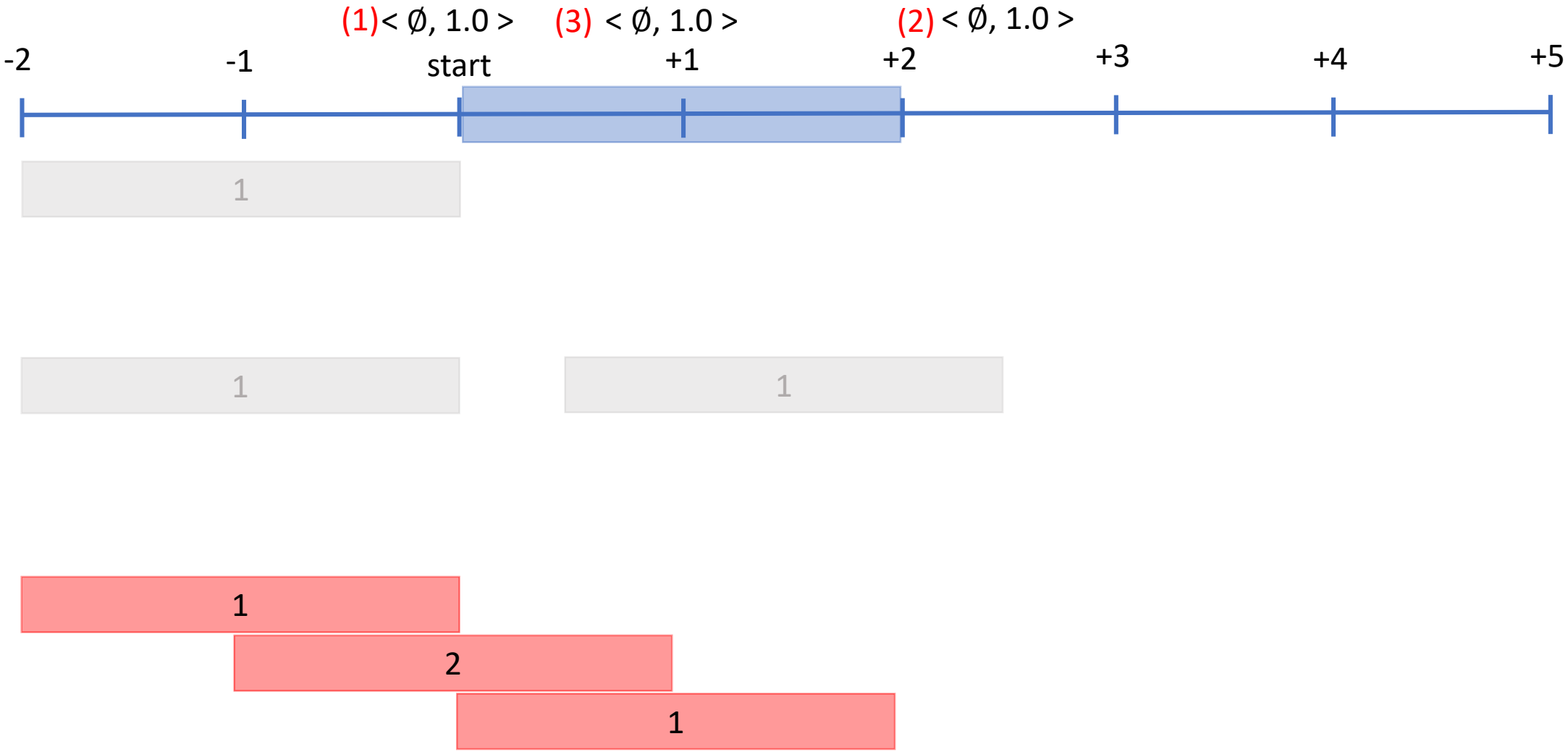
Bonus #1: windowing – exemple jouet – Sliding window



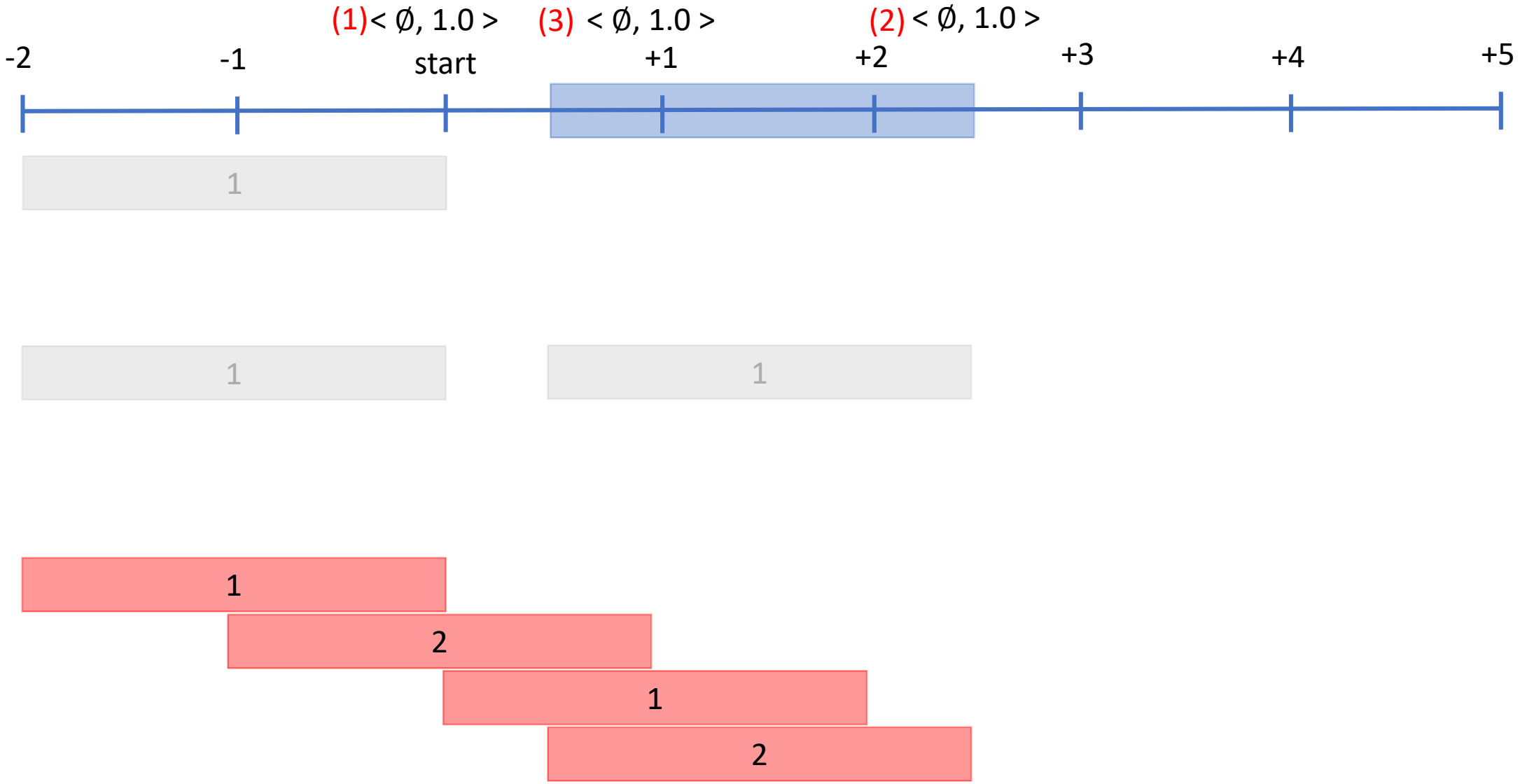
Bonus #1: windowing – exemple jouet – Sliding window



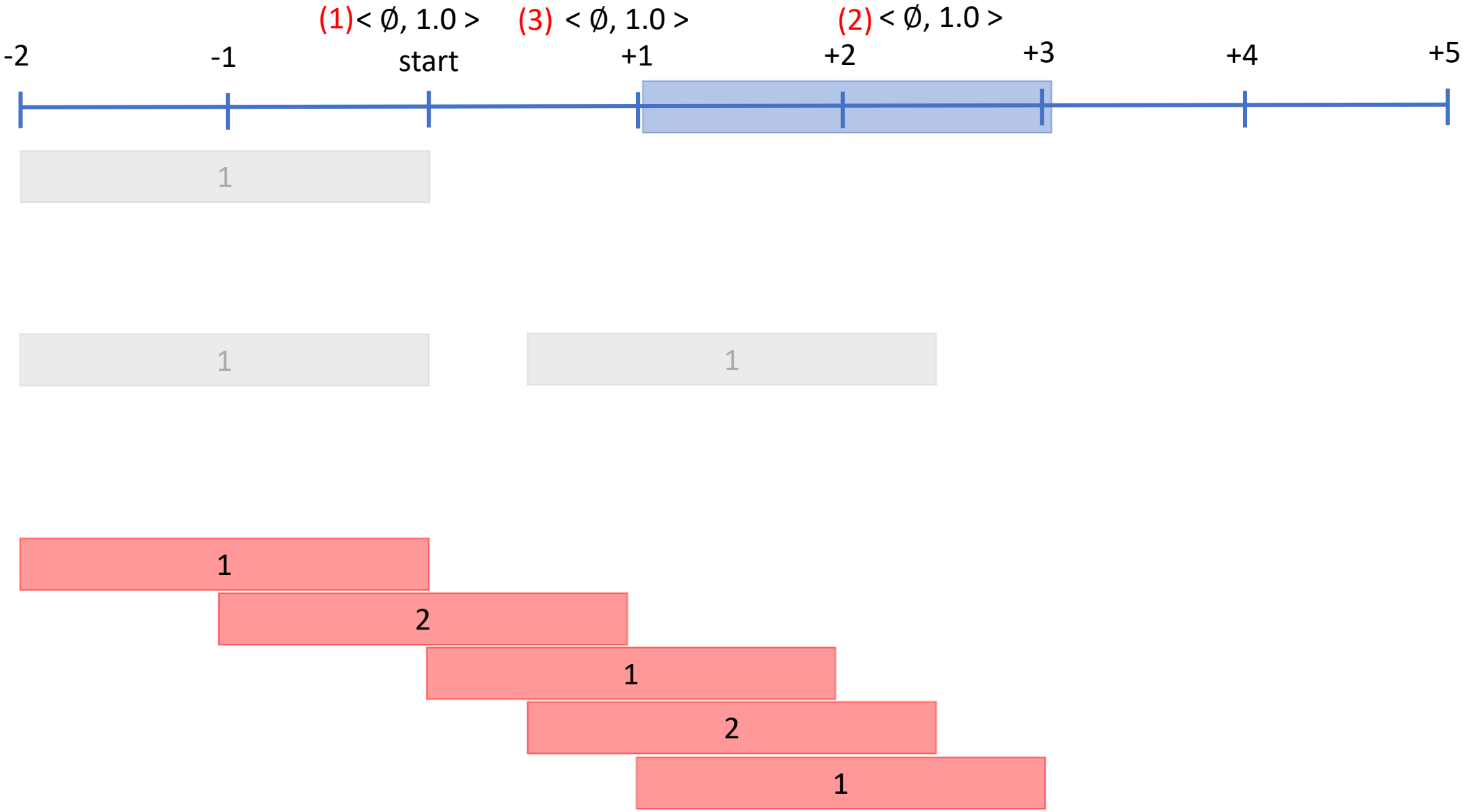
Bonus #1: windowing – exemple jouet – Sliding window



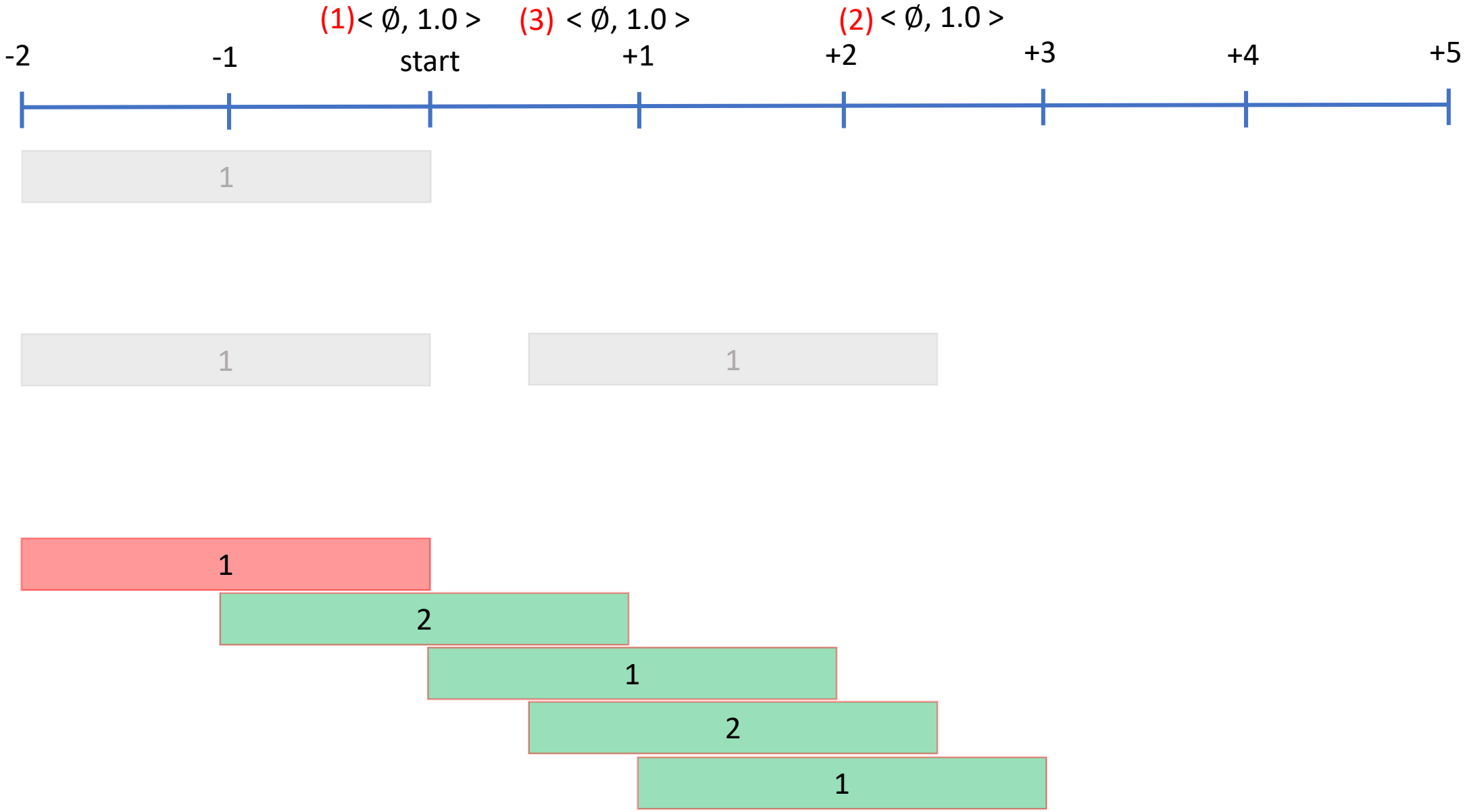
Bonus #1: windowing – exemple jouet – Sliding window



Bonus #1: windowing – exemple jouet – Sliding window



Bonus #1: windowing – exemple jouet – Sliding window



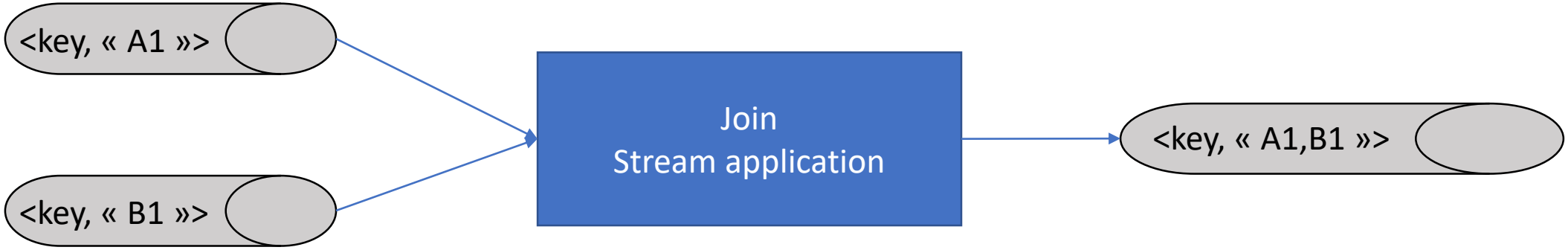
Bonus #1: windowing - Les différents types de windowing

- Session Windows
- Hopping Windows
- Sliding Windows
- Timbling Windows

Bonus #2: joins - Sommaire

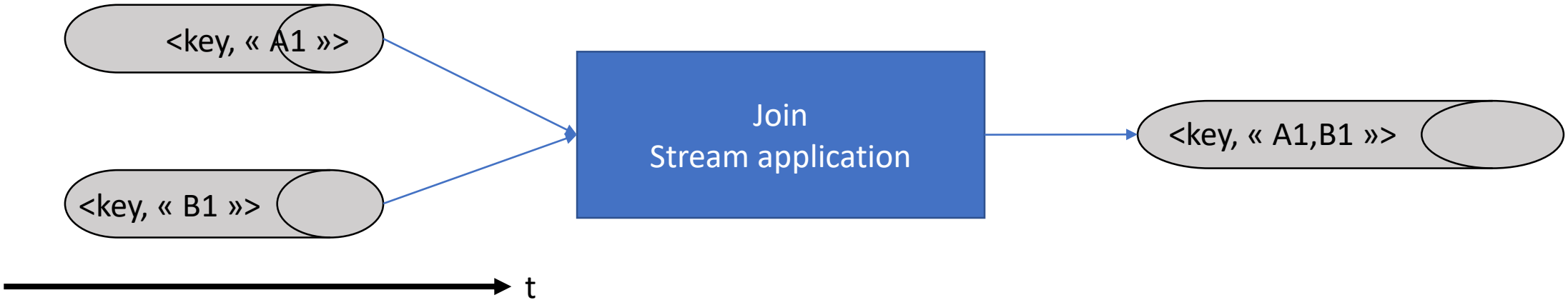
- Un premier exemple jouet
- Les différents types de joins

Bonus #2: joins - Un premier exemple jouet



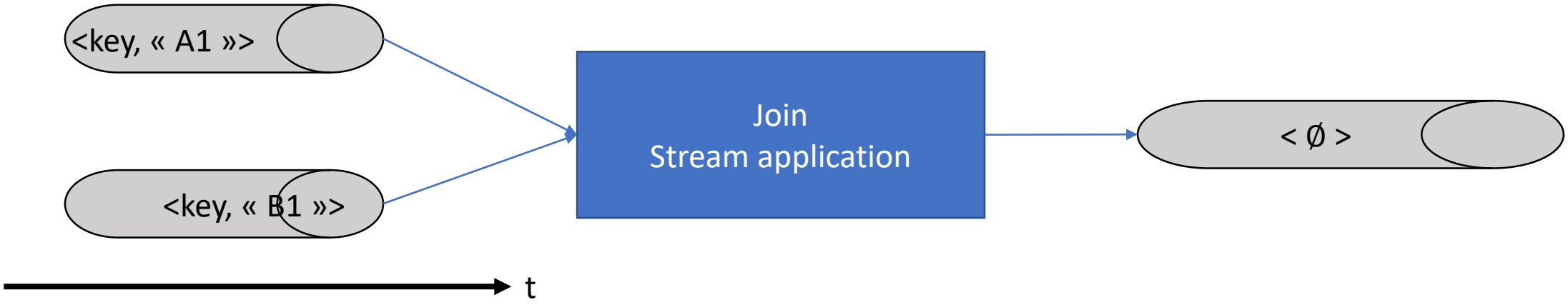

```
public static Topology getTopology() {
  StreamsBuilder builder = new StreamsBuilder();
  KStream<String, String> inputAStream = builder.stream(INPUT_TOPIC_A);
  KStream<String, String> inputBStream = builder.stream(INPUT_TOPIC_B);
  inputAStream
      .join(
          inputBStream.toTable(),
          (value1, value2) -> String.join(",", value1, value2))
      .to(OUTPUT_TOPIC);
  return builder.build();
}
```

Bonus #2: joins - Un premier exemple jouet




```
public static Topology getTopology() {
  StreamsBuilder builder = new StreamsBuilder();
  KStream<String, String> inputAStream = builder.stream(INPUT_TOPIC_A);
  KStream<String, String> inputBStream = builder.stream(INPUT_TOPIC_B);
  inputAStream
      .join(
          inputBStream.toTable(),
          (value1, value2) -> String.join(",", value1, value2))
      .to(OUTPUT_TOPIC);
  return builder.build();
}
```

Bonus #2: joins - Un premier exemple jouet




```
public static Topology getTopology() {
  StreamsBuilder builder = new StreamsBuilder();
  KStream<String, String> inputAStream = builder.stream(INPUT_TOPIC_A);
  KStream<String, String> inputBStream = builder.stream(INPUT_TOPIC_B);
  inputAStream
      .join(
          inputBStream.toTable(),
          (value1, value2) -> String.join(",", value1, value2))
      .to(OUTPUT_TOPIC);
  return builder.build();
}
```

Conclusions ?

Conclusions ?

- Une courbe d’apprentissage plus importante qu’anticipé …

Conclusions ?

- Une courbe d’apprentissage plus importante qu’anticipé …
  - … Mais je suis convaincu que c’est rentable

Conclusions ?

- Une courbe d’apprentissage plus importante qu’anticipé …
  - … Mais je suis convaincu que c’est rentable
- L’application crée automatiquement des topics (repartition, changelog / statestore) …

Conclusions ?

- Une courbe d’apprentissage plus importante qu’anticipé …
  - … Mais je suis convaincu que c’est rentable
- L’application crée automatiquement des topics (repartition, changelog / statestore) …
  - … Est-ce que ça marcherait si la création automatique est désactivée (et c’est le cas de l’EventBus) ?

Conclusions ?

- Une courbe d’apprentissage plus importante qu’anticipé …
  - … Mais je suis convaincu que c’est rentable
- L’application crée automatiquement des topics (repartition, changelog / statestore) …
  - … Est-ce que ça marcherait si la création automatique est désactivée (et c’est le cas de l’EventBus) ?
  - Next step : essayer de faire tourner l’exemple « wordcount » sur l’EventBus

Conclusions ?

- Une courbe d'apprentissage plus importante qu'anticipé …
  - … Mais je suis convaincu que c'est rentable
- L'application crée automatiquement des topics (repartition, changelog / statestore) …
  - … Est-ce que ça marcherait si la création automatique est désactivée (et c'est le cas de l'EventBus) ?
  - Next step : essayer de faire tourner l'exemple « wordcount » sur l'EventBus
- Une ou plusieurs suite … si le temps le permet

Conclusions ?

- Une courbe d'apprentissage plus importante qu'anticipé …
  - … Mais je suis convaincu que c'est rentable
- L'application crée automatiquement des topics (repartition, changelog / statestore) …
  - … Est-ce que ça marcherait si la création automatique est désactivée (et c'est le cas de l'EventBus) ?
  - Next step : essayer de faire tourner l'exemple « wordcount » sur l'EventBus
- Une ou plusieurs suite … si le temps le permet
  - Mieux présenter les différents types de windowing et de joins

Conclusions ?

- Une courbe d'apprentissage plus importante qu'anticipé …
  - … Mais je suis convaincu que c'est rentable
- L'application crée automatiquement des topics (repartition, changelog / statestore) …
  - … Est-ce que ça marcherait si la création automatique est désactivée (et c'est le cas de l'EventBus) ?
  - Next step : essayer de faire tourner l'exemple « wordcount » sur l'EventBus
- Une ou plusieurs suite … si le temps le permet
  - Mieux présenter les différents types de windowing et de joins
  - Des exemples plus « cas réels » (clickstreams, possibles cas ING, etc.)

Conclusions ?

- Une courbe d'apprentissage plus importante qu'anticipé …
  - … Mais je suis convaincu que c'est rentable
- L'application crée automatiquement des topics (repartition, changelog / statestore) …
  - … Est-ce que ça marcherait si la création automatique est désactivée (et c'est le cas de l'EventBus) ?
  - Next step : essayer de faire tourner l'exemple « wordcount » sur l'EventBus
- Une ou plusieurs suite … si le temps le permet
  - Mieux présenter les différents types de windowing et de joins
  - Des exemples plus « cas réels » (clickstreams, possibles cas ING, etc.)
  - Exemples de déploiements

Conclusions ?

- Une courbe d'apprentissage plus importante qu'anticipé …
  - … Mais je suis convaincu que c'est rentable
- L'application crée automatiquement des topics (repartition, changelog / statestore) …
  - … Est-ce que ça marcherait si la création automatique est désactivée (et c'est le cas de l'EventBus) ?
  - Next step : essayer de faire tourner l'exemple « wordcount » sur l'EventBus
- Une ou plusieurs suite … si le temps le permet
  - Mieux présenter les différents types de windowing et de joins
  - Des exemples plus « cas réels » (clickstreams, possibles cas ING, etc.)
  - Exemples de déploiements
  - KsqlDB -> un langage proche de SQL, surcouche à kafka-streams