

Gestion de Grandes Masses de Données

GGMD

NICOLAS LUMINEAU



département
Informatique

Université Claude Bernard Lyon 1

PARCOURS TIW / DS-INFO / BIO-INFO

nicolas.lumineau@univ-lyon1.fr

Big Data

Volume

Vélocité

Variété

Véracité

Valeur

Evolution des besoins

A partir des données stockées en base, donner le nombre de jours en mai 2023 où le niveau d'alerte 1 a été déclenché pour une pollution aux particules PM10.

A partir de 3 capteurs se trouvant à Villeurbanne, retourner les valeurs de Dioxyde d'azote supérieures à $40\mu\text{g}/\text{m}^3$ observées.

Donner la moyenne sur les 15 dernières minutes du niveau de polluants HAP observées à partir des 10 000 capteurs répartis dans la région Auvergne-Rhône-Alpes et ce, toutes les 5 minutes.

Contexte

Système de Gestion des Flux de Données (Data Stream Management System DSMS)

Traitement de flux de données à débit variant

- ▶ Contraintes:
 - ▶ Maîtriser la qualité des résultats retournés
 - ▶ Maîtriser les ressources (CPU, RAM bande passante) nécessaires au traitement

Plan

Concepts et définitions

- Flux de données
- Requêtes continues
 - Opérateurs
 - Fenêtrage
- Infrastructure d'exécution

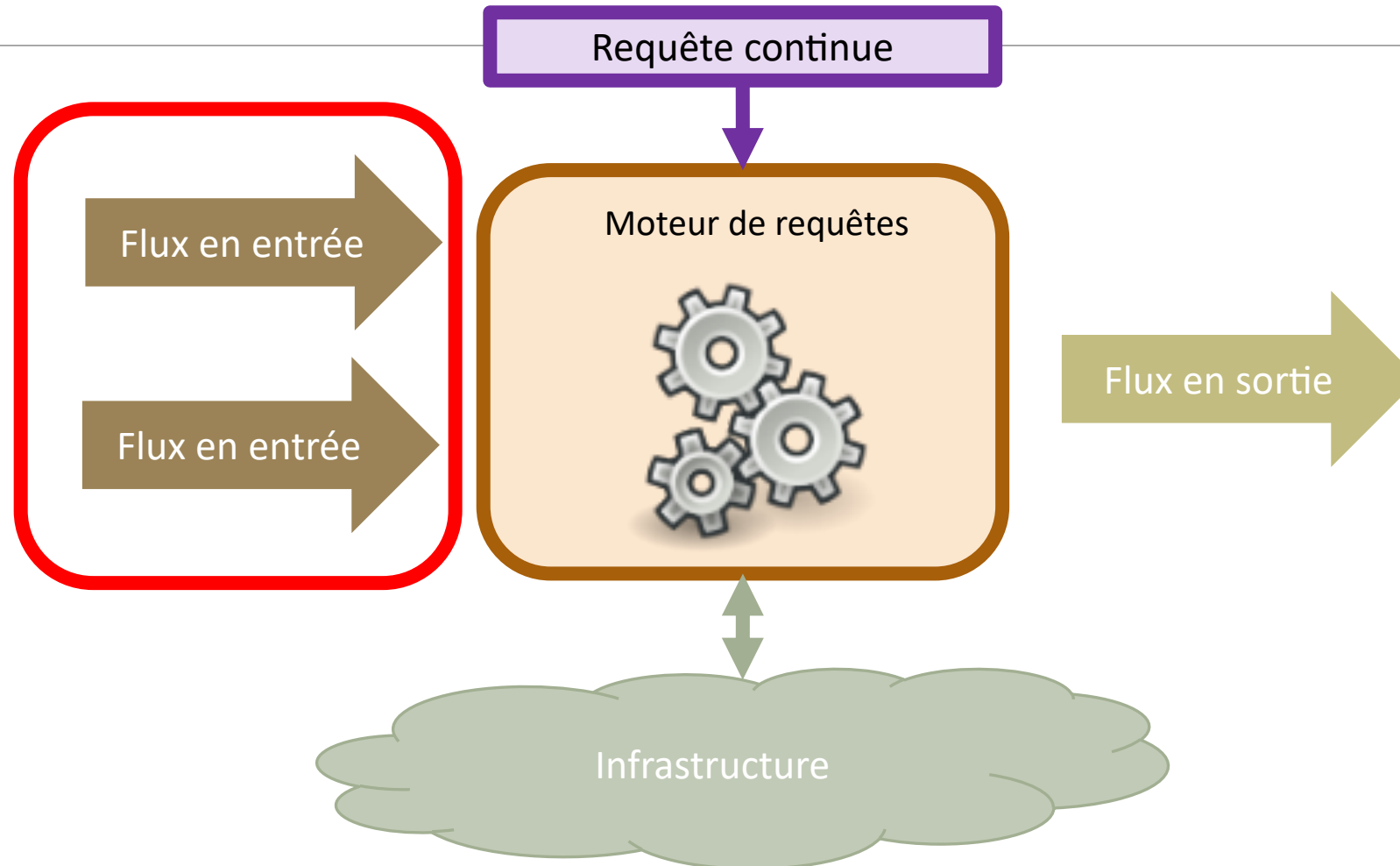
Systèmes de gestion de flux (DSMS)

- Challenges et cas critiques
- Congestion
- Elasticité
- Classifications des DSMS

Approches 'Workflow'

- Choix d'optimisation

Principe du traitement de flux



Flux de données : définition

Un flux S est un ensemble potentiellement infini de paires $\langle s, \tau \rangle$, où :

- s est un tuple respectant le schéma de S
- τ est le *timestamp* de cet élément.

Remarque :

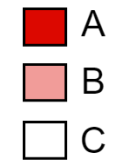
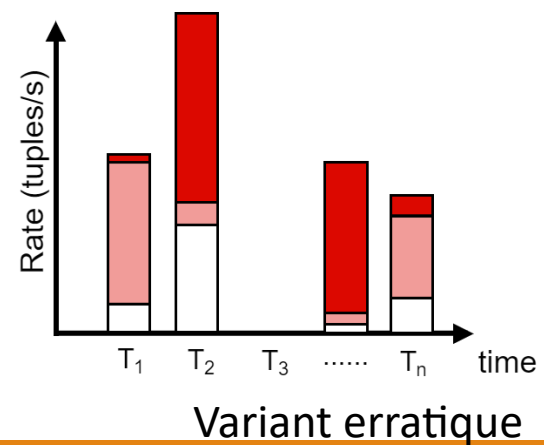
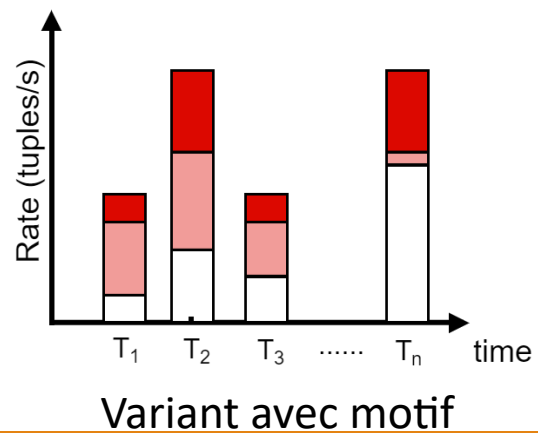
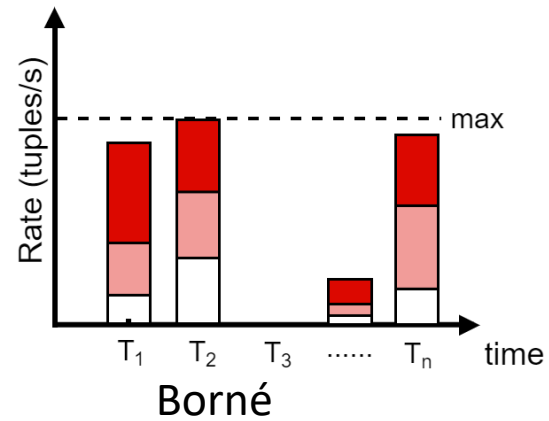
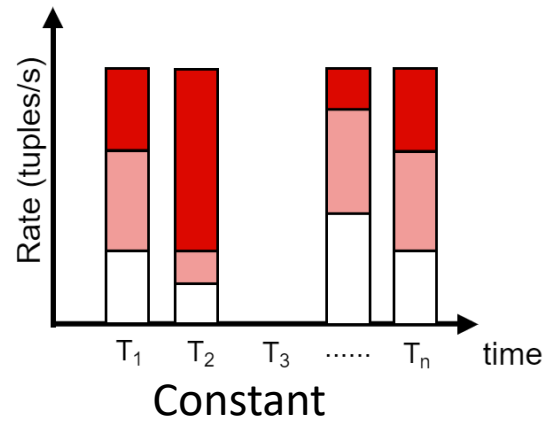
Un *timestamp* τ n'appartient pas au schéma de S et il peut y avoir zéro, un ou plusieurs éléments de S partageant le même τ .

Flux de données : caractéristiques

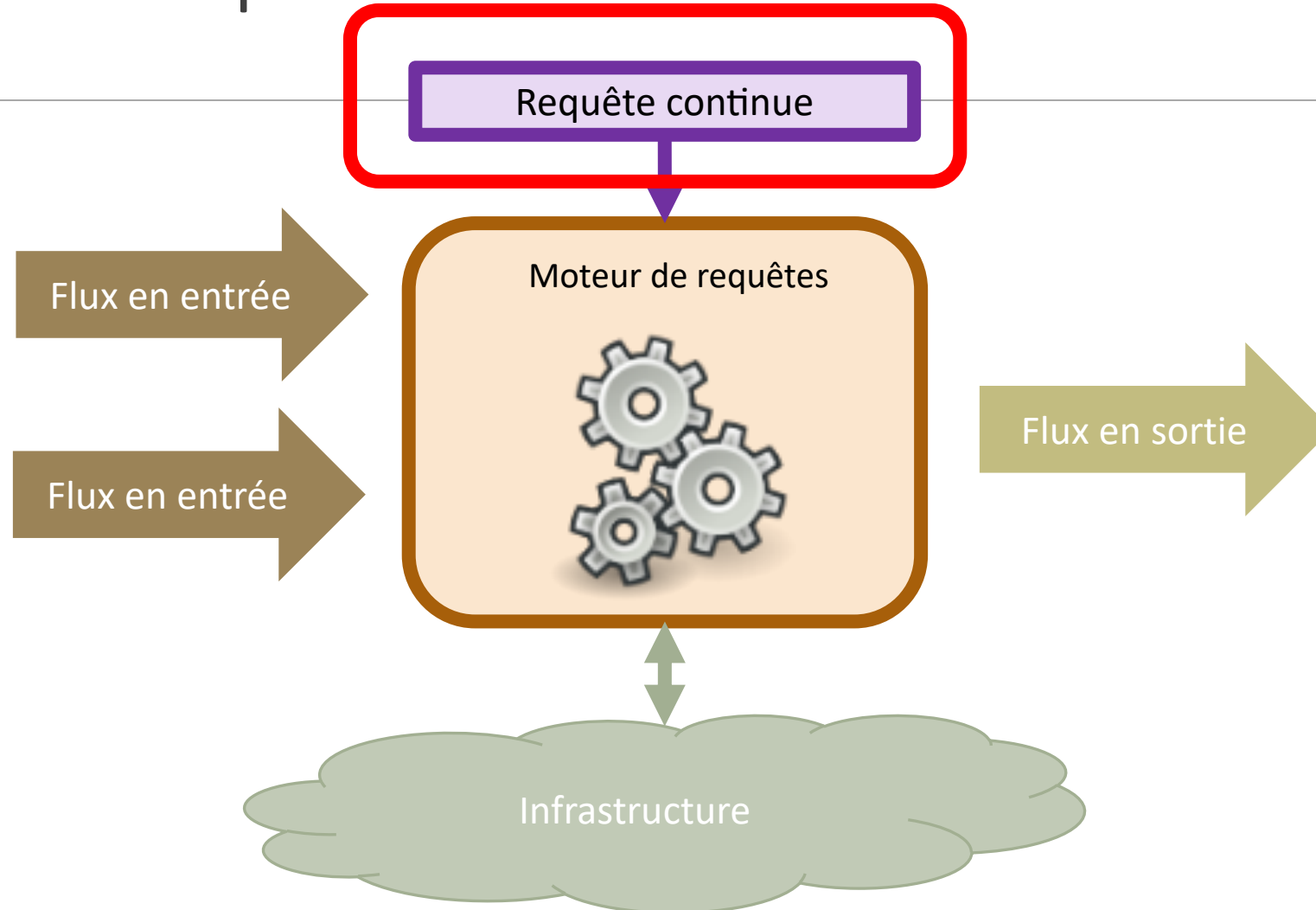
Un flux S est caractérisé par:

- La valeur de son débit
 - exprimé en nombre de tuples par unité de temps
- L'évolution de son débit dans le temps
 - Débit constant
 - Débit borné
 - Débit variant
 - Avec motifs régulier
 - De manière erratique
- La distribution de ses valeurs
 - exprimée selon une loi de distribution
 - Uniforme
 - Zipf
 - ...

Flux de données : exemples



Principe du traitement de flux



Requêtes continues

- Une requête continue Q est une requête qui est émise une fois et qui s'exécute de façon logique sur les données jusqu'à ce que Q soit terminée.
- Le traitement d'une requête continue nécessite l'exécution d'opérateurs (avec ou sans état) avec une fréquence et une portée paramétrées.

Opérateurs

Le traitement d'une requête continue peut nécessiter des opérateurs de différents types :

Opérateur sans état ('Stateless')

- Calcul indépendant de l'état/résultat calculé précédemment
- Exemple: requête appliquant un filtre sur les données

Opérateur avec état ('Stateful')

- Calcul dépendant de l'état/résultat calculé précédemment
- Exemple : requête calculant l'évolution du nombre moyen de valeurs apparaissant toutes les δ secondes

Requêtes continues : paramètres

Une requête continue dispose :

- d'une fréquence de résultat attendu

*Donner la moyenne sur les 15 dernières minutes du niveau de polluants HAP observées à partir des 10 000 capteurs répartis dans la région Auvergne-Rhône-Alpes et ce **toutes les 5 minutes**.*

- d'une portée sur laquelle la requête est calculée

*Donner la moyenne **sur les 15 dernières minutes** du niveau de polluants HAP observées à partir des 10 000 capteurs répartis dans la région Auvergne-Rhône-Alpes et ce toutes les 5 minutes.*

Fenêtrage

Rappel:

- Un flux est infini

Conséquence:

- Il n'est pas possible d'interroger le flux dans sa globalité

Besoin:

- Système de fenêtrage

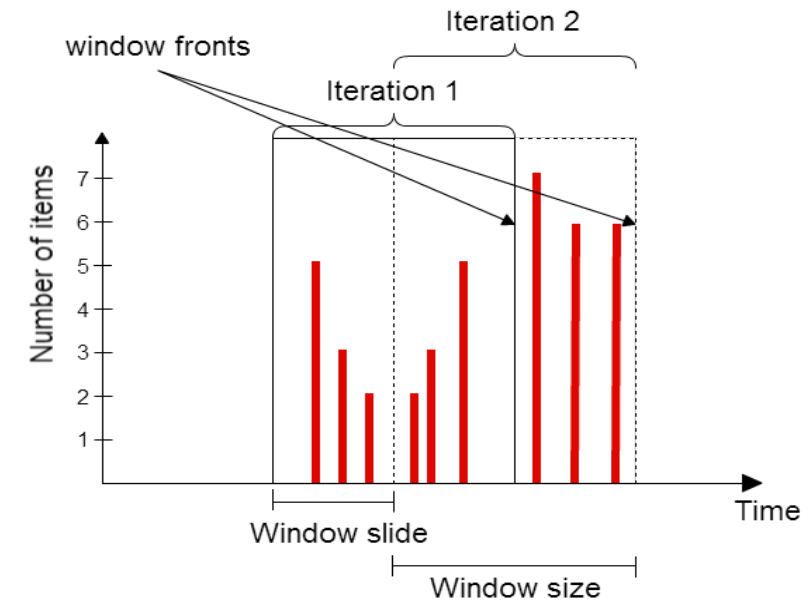
Fenêtrage

Fenêtres glissantes

- Paramètres: taille, pas
- Différents types
 - Fenêtres basées sur le temps
 - Fenêtres basées sur le nombre de tuples reçus
 - Fenêtres basées sur le nombre de valeurs de tuples reçus

Fenêtres sautantes

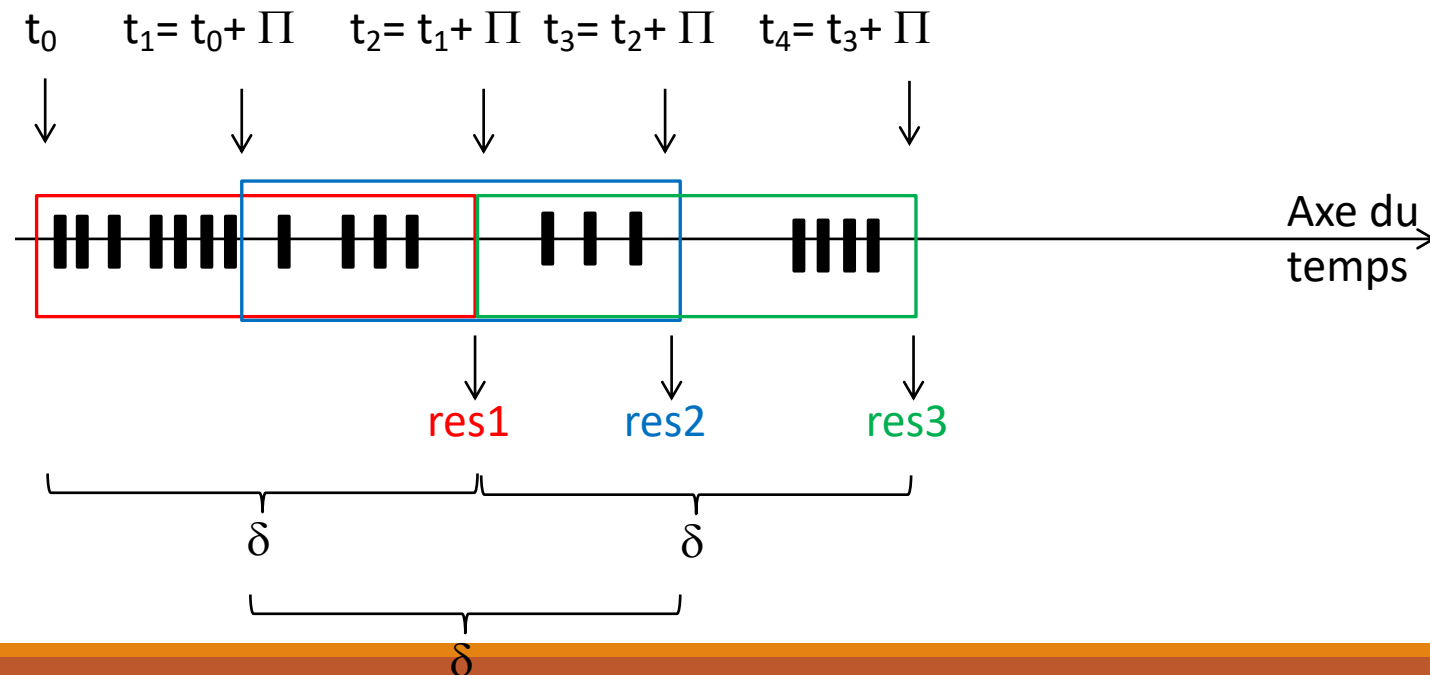
- Fenêtre glissante où pas=taille



Fenêtrage

Fenêtre basée sur le temps (Time-based)

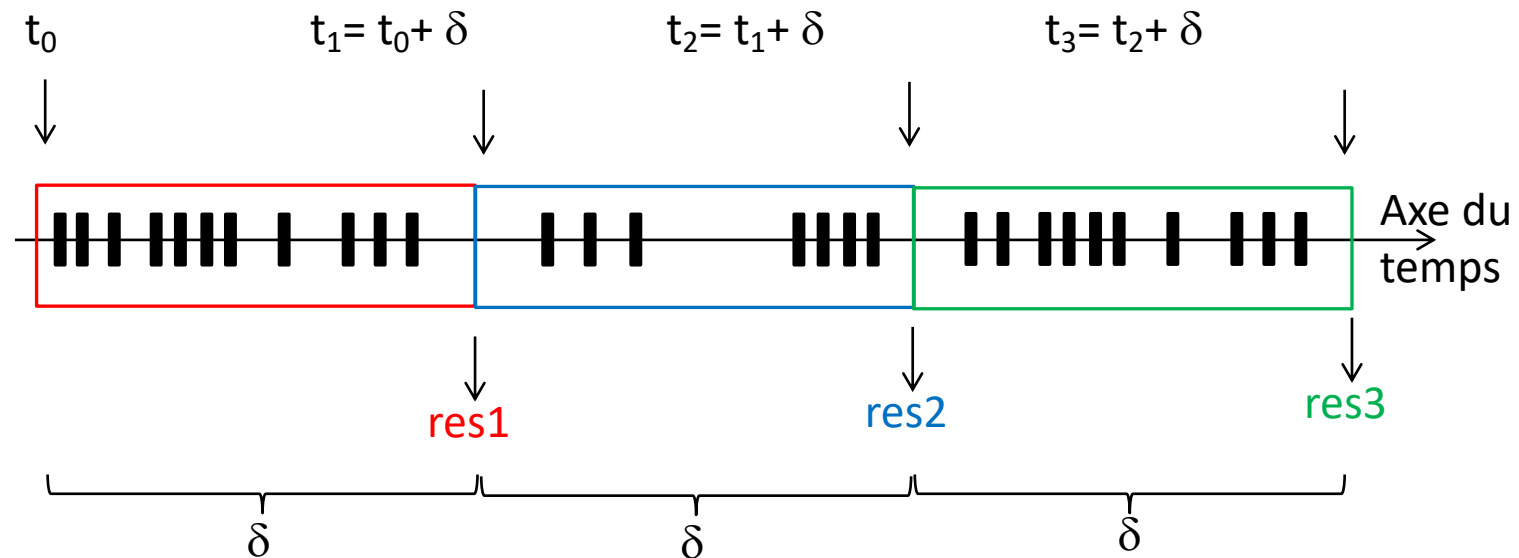
- La fenêtre est définie à partir d'une durée
- Le calcul de la requête se fait sur les tuples reçus depuis δ secondes et ce, toutes les Π secondes



Fenêtrage

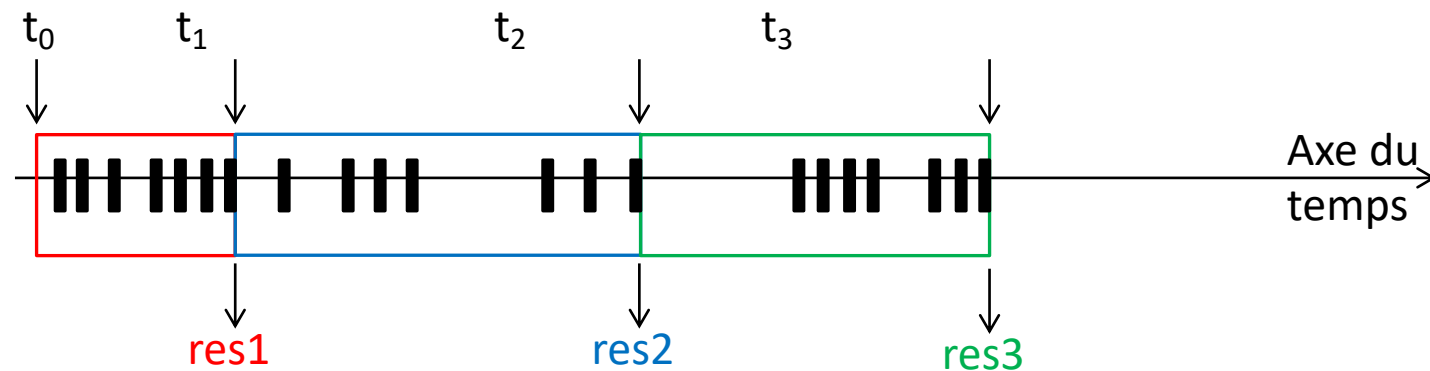
Cas de la fenêtre sautante:

- La fenêtre est définie à partir d'une durée
- Le calcul de la requête se fait sur les tuples reçus depuis δ secondes et ce, toutes les δ secondes



Fenêtrage

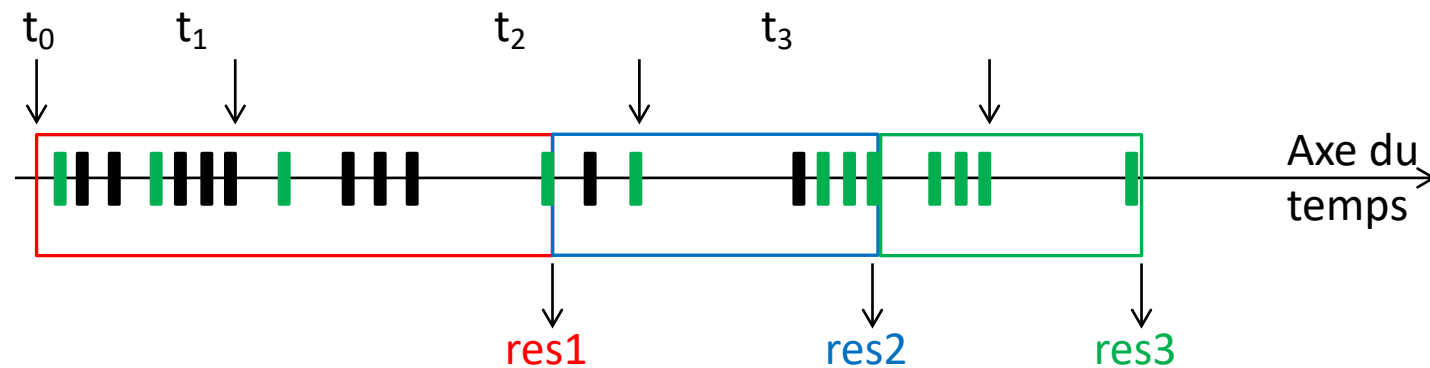
- Fenêtres basées sur le nombre de tuples reçus (Tuple-based)
 - La fenêtre est définie à partir d'un nombre de tuples reçus
 - Le calcul de la requête se fait sur les n tuples reçus



$n = 7$

Fenêtrage

- Fenêtres basées sur le nombre de tuples reçus (Partition-based)
- La fenêtre est définie à partir d'un nombre de valeurs de tuples reçus
 - Le calcul de la requête se fait sur les n tuples reçus et vérifiant un formule logique



$n = 4$

Gestion du temps

Problème de synchronisation présent en cas de plusieurs sources (ordre non garanti, latence réseau...).

Système en 'battement de cœur' possible de deux manière :

- associer un timestamp à chaque tuple entrant.
- chaque source envoie une ponctuation lorsqu'elle a fini d'envoyer des tuples associées à un timestamp et la synchronisation s'effectue grâce aux sources.

Continuous Query Language (CQL)

Langage déclaratif basé sur SQL pour l'écriture de requêtes continues sur des flux ou des jeux de données évoluant périodiquement.

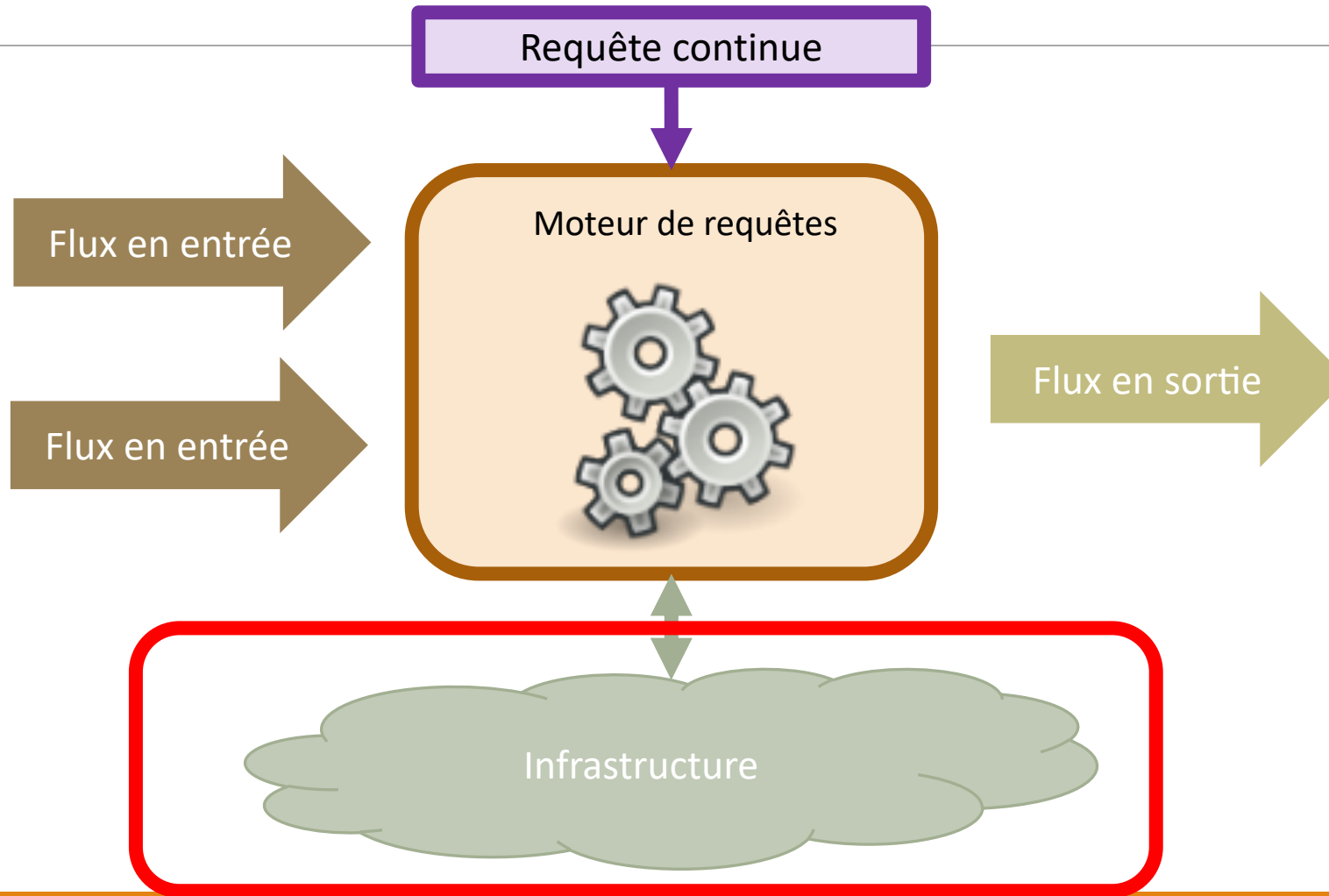
CQL intègre une définition des fenêtres glissantes dérivée de SQL-99.

Trois classes d'opérateurs en CQL :

- **Stream-to-relation** : opérateurs exprimés grâce aux fenêtres glissantes (dérivées de SQL-99).
- **Relation-to-relation** : opérateurs issus de SQL.
- **Relation-to-stream** : trois opérateurs spécifiques : Istream (Insert Stream), Dstream (Delete Stream), Rstream (Relation Stream)

Arasu, Arvind & Babu, Shivnath & Widom, Jennifer. (2004). The CQL Continuous Query Language: Semantic Foundations and Query Execution. VLDB J

Principe du traitement de flux



Infrastructure

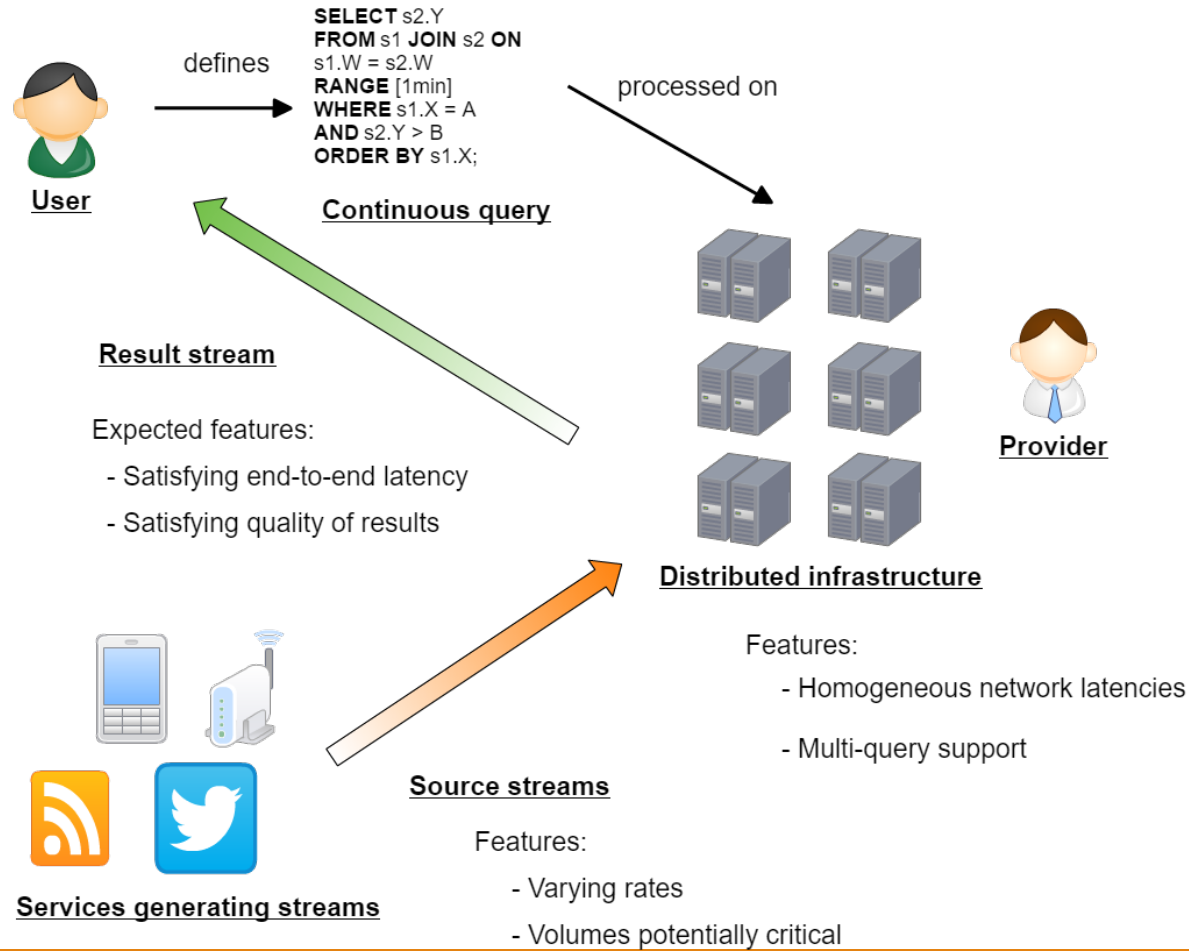
Centralisée

- multicœurs
 - Possibilité de parallélisation des *threads*

Distribuée

- Grille, Cloud
 - Possibilité de parallélisation des *threads*
 - Mise à disposition de nouvelles ressources
 - Introduction de latence réseau

Contexte Big Data



Plan

Concepts et définitions

- Flux de données
- Requêtes continues
 - Opérateurs
 - Fenêtrage
- Infrastructure d'exécution

Systèmes de gestion de flux (DSMS)

- Challenges et cas critiques
- Congestion
- Elasticité
- Classifications des DSMS

Approches 'Workflow'

- Choix d'optimisation

Challenges

Performance de traitement

- Pourvoir obtenir des résultats en quasi-temps réel
- Optimiser le traitement de la requête

Qualité des résultats

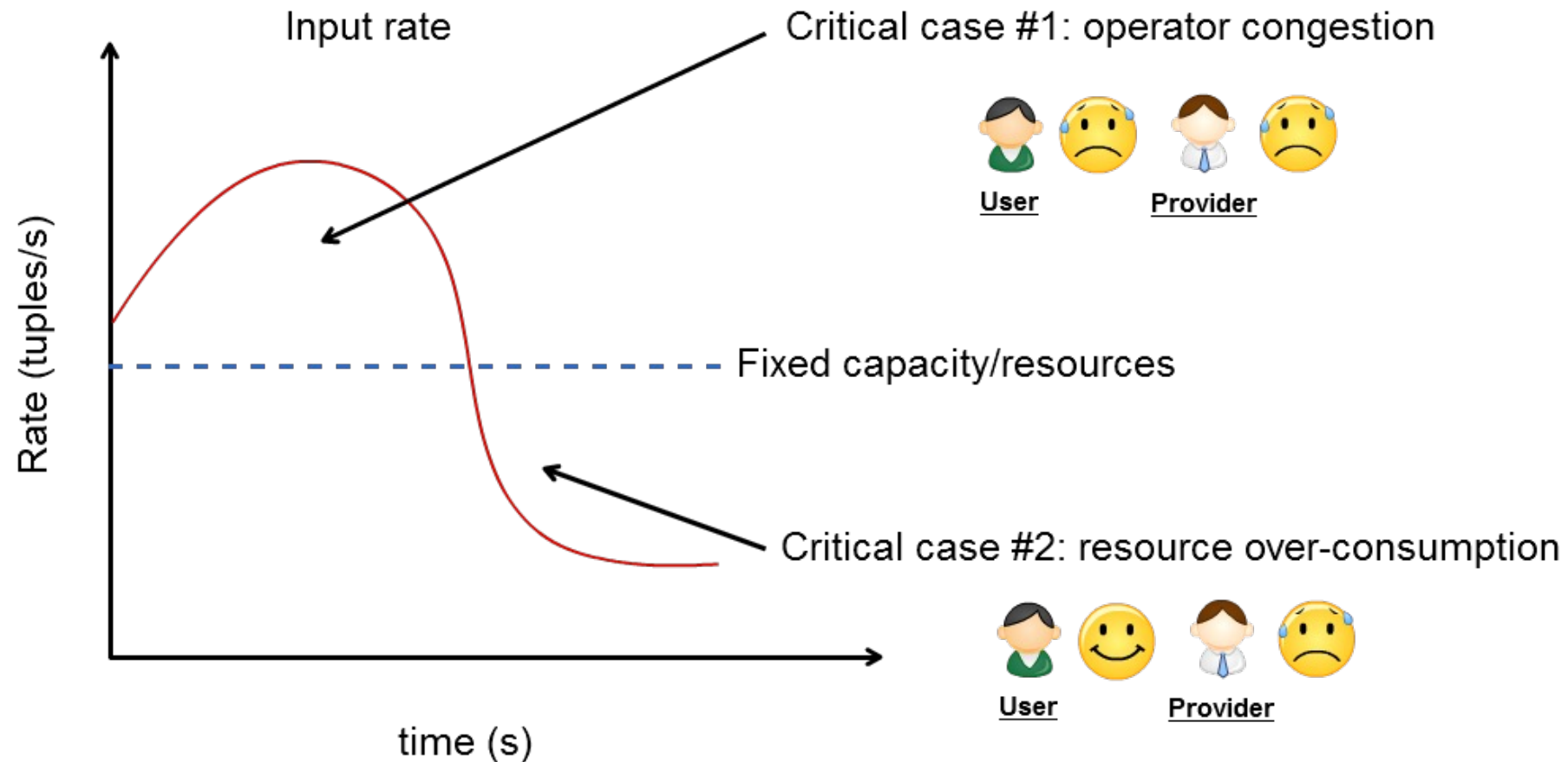
- Obtenir des résultats en ayant limité les pertes de données
- Eviter la congestion des opérateurs

Adaptabilité des ressources

- Avoir les ressources nécessaires pour traiter le flux
- Ne pas bloquer des ressources inutilement




Cas critiques

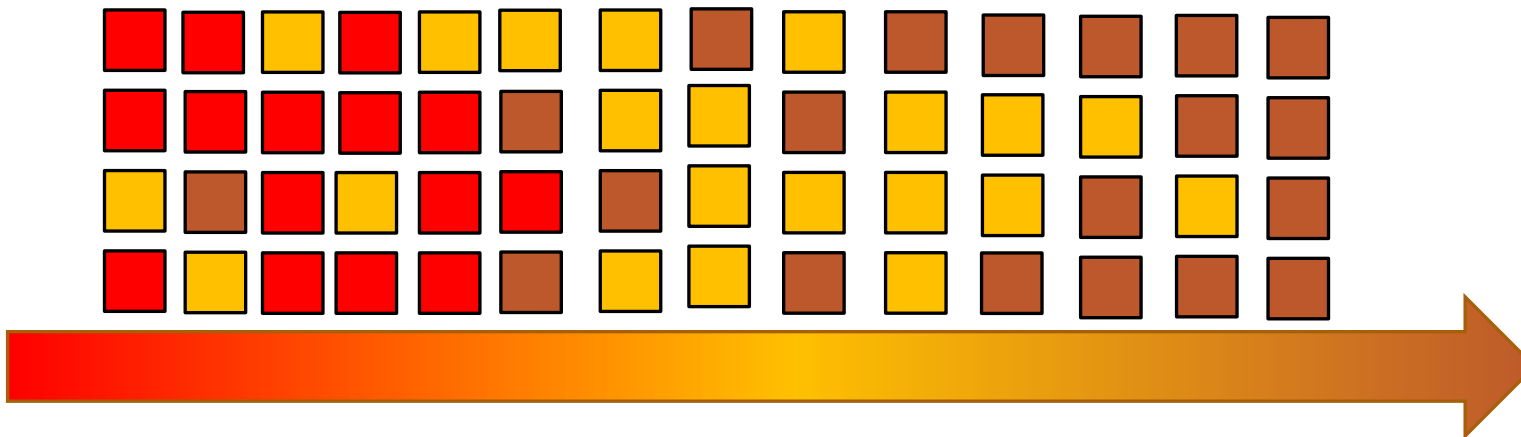
Evolution du débit des flux



Cas critiques

Evolution de la distribution des valeurs dans le flux

-  A Temps de traitement moyen d'un A : X ms
-  B Temps de traitement moyen d'un B : 100.X ms
-  C Temps de traitement moyen d'un C : 1000.X ms

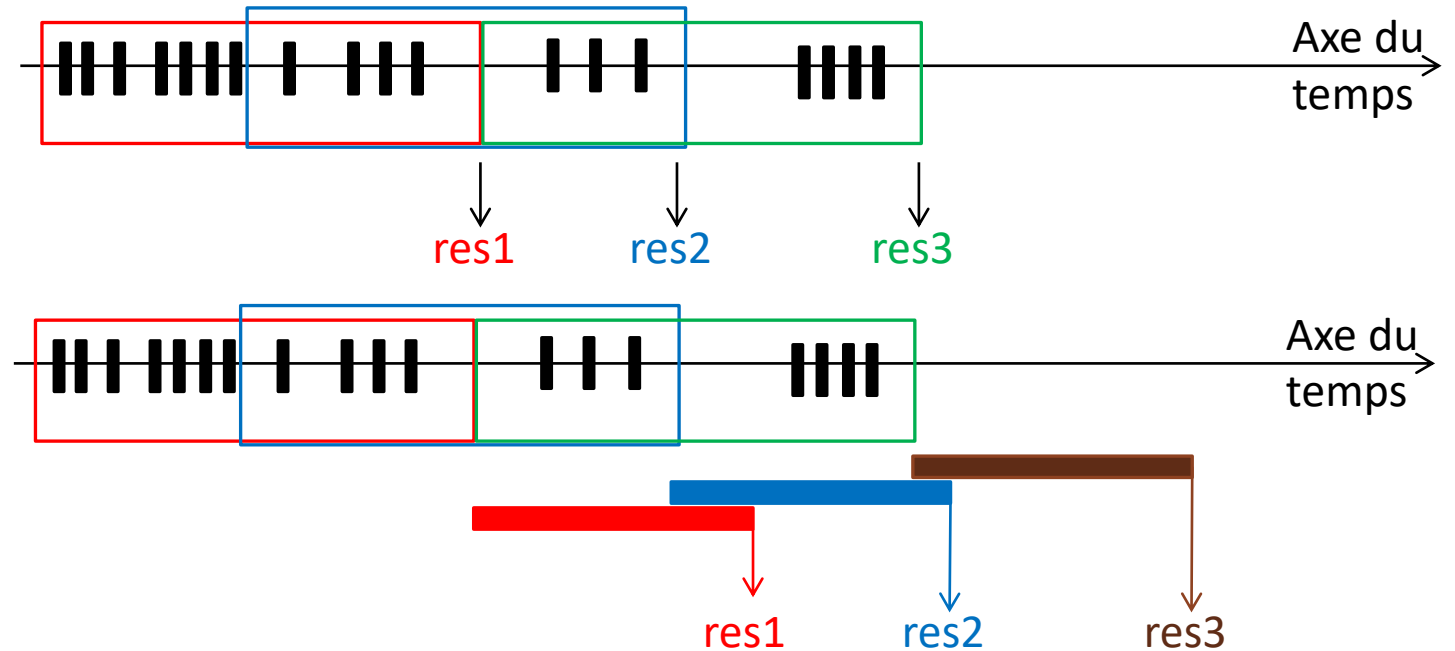


Remarque

La fin d'une fenêtre n'implique pas forcément la disponibilité d'un résultat.

Exemple:

- Calcul instantané du résultat
- Calcul couteux du résultat



Congestion

Le système est considéré comme congestionné, si :

- des données en entrée du système sont perdues (file d'attente saturée)
- ou
- des données avec TTL dépassé

Elasticité des DSMS

Définition

- Adapter les ressources nécessaires au traitement des requêtes

Enjeux (intersection Big Data / Green IT)

- Performance du traitement de la requête
- Qualité des résultats retournés
- Consommation énergétique
 - Pouvoir adapter la quantité de ressources nécessaires en cas de variations du débit des flux à traiter
- Automaticité
 - Pouvoir adapter automatiquement les ressources

Gestion de l'élasticité

Deux niveaux de traitements de l'élasticité

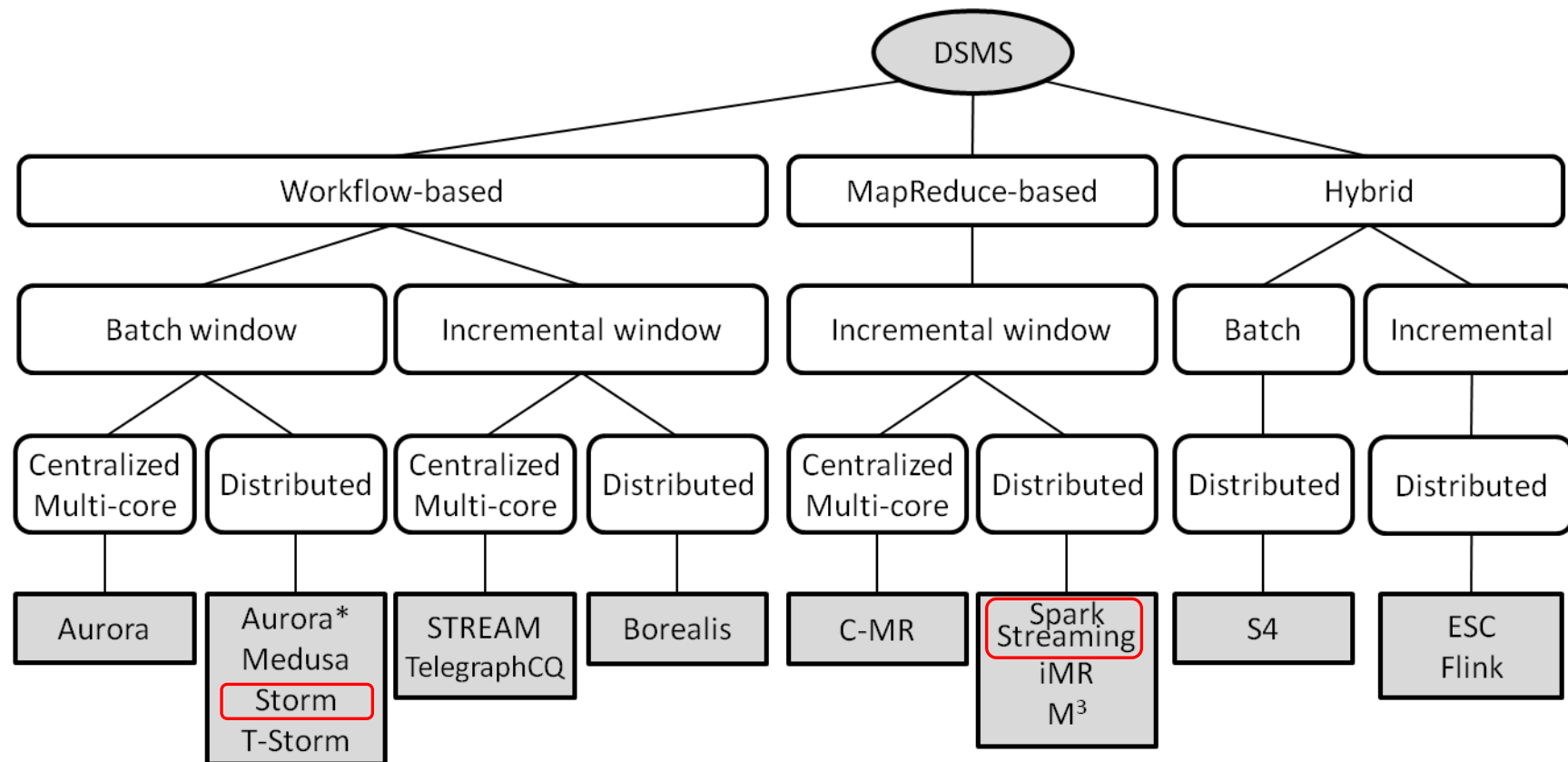
- Ajout/suppression de ressources supplémentaires pour l'exécution des *threads*
- Concentration/Etalement des threads

Deux contextes :

- Ressources disponibles et suffisantes
- Ressources limitées et insuffisantes

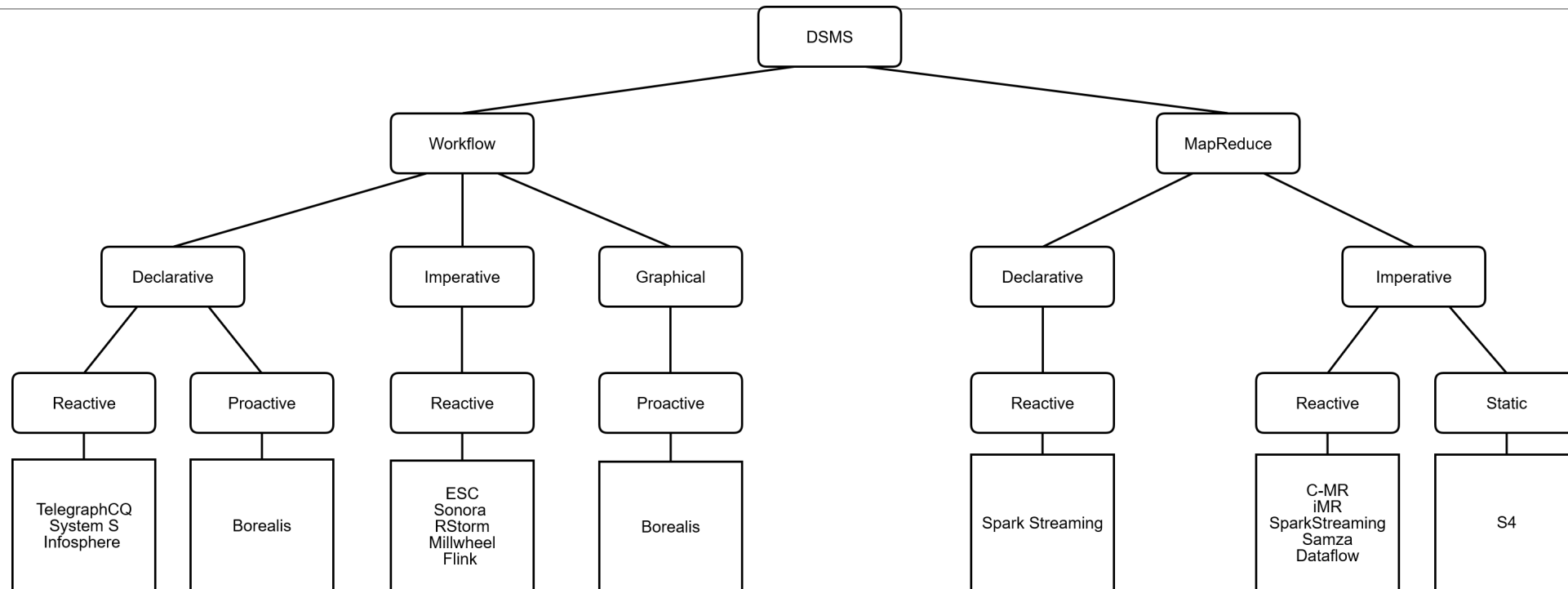
Classification des DSMS parallèles

(Paradigme/Traitement/Support)



Classification des DSMS parallèles

(Paradigme/langage/gestion congestion)



Map Reduce vs Workflow



Plan

Concepts et définitions

- Flux de données
- Requêtes continues
 - Opérateurs
 - Fenêtrage
- Infrastructure d'exécution

Systèmes de gestion de flux (DSMS)

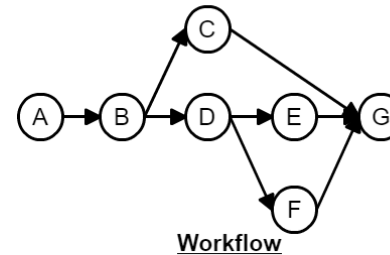
- Challenges et cas critiques
- Congestion
- Elasticité
- Classifications des DSMS

Approches 'Workflow'

- Choix d'optimisation

Approches Workflow

Les requêtes sont considérées
comme des graphes d'opérateurs



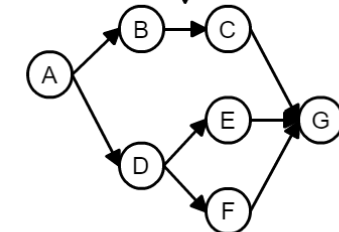
```
SELECT s2.Y  
FROM s1 JOIN s2 ON  
s1.W = s2.W  
RANGE [1min]  
WHERE s1.X = A  
AND s2.Y > B  
ORDER BY s1.X;
```

Continuous query

submitted to

**Workflow-based
stream processing engine**

generates



processed on

Distributed infrastructure



Execution context

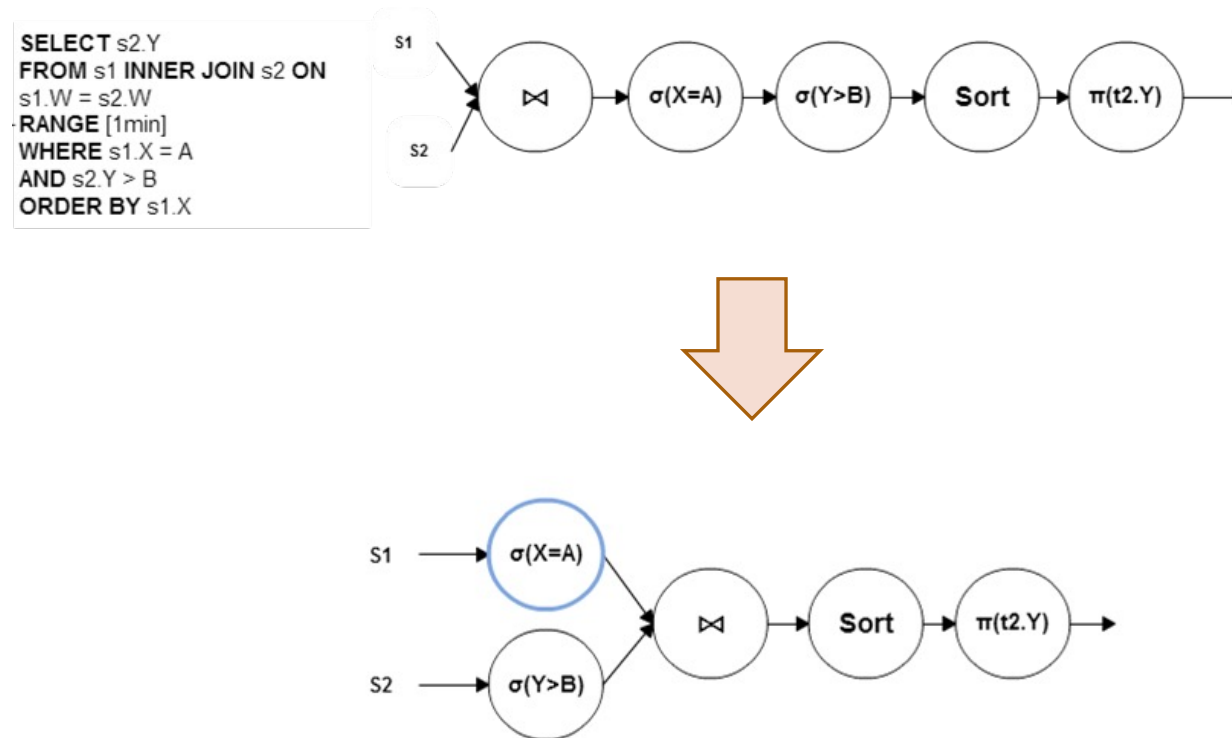
Traitement de requêtes

Des choix d'optimisation :

- Choix de la topologie
 - Définition d'une topologie réduisant le nombre de tuples manipulés
- Choix du degré de parallélisme des opérateurs
 - Définition du nombre de threads permettant de paralléliser l'exécution d'un opérateur
- Choix de l'allocation d'une tâche sur une unité de traitement
 - Définition d'une stratégie d'allocation des threads sur les unités de traitement
- Choix de l'implémentation des opérateurs
 - Définition d'une stratégie de sélection d'implémentation des opérateurs

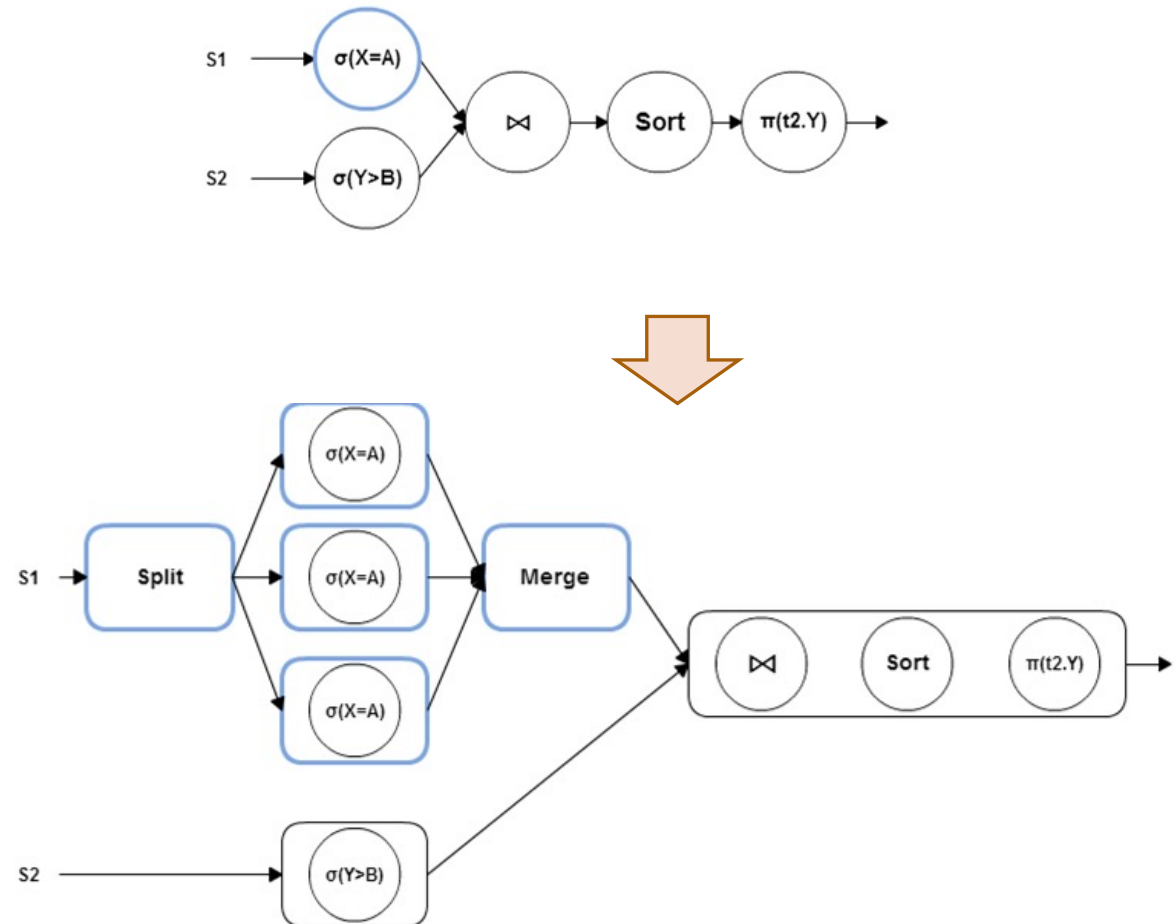
Choix de la topologie

Niveau logique



Choix du degré de parallélisme

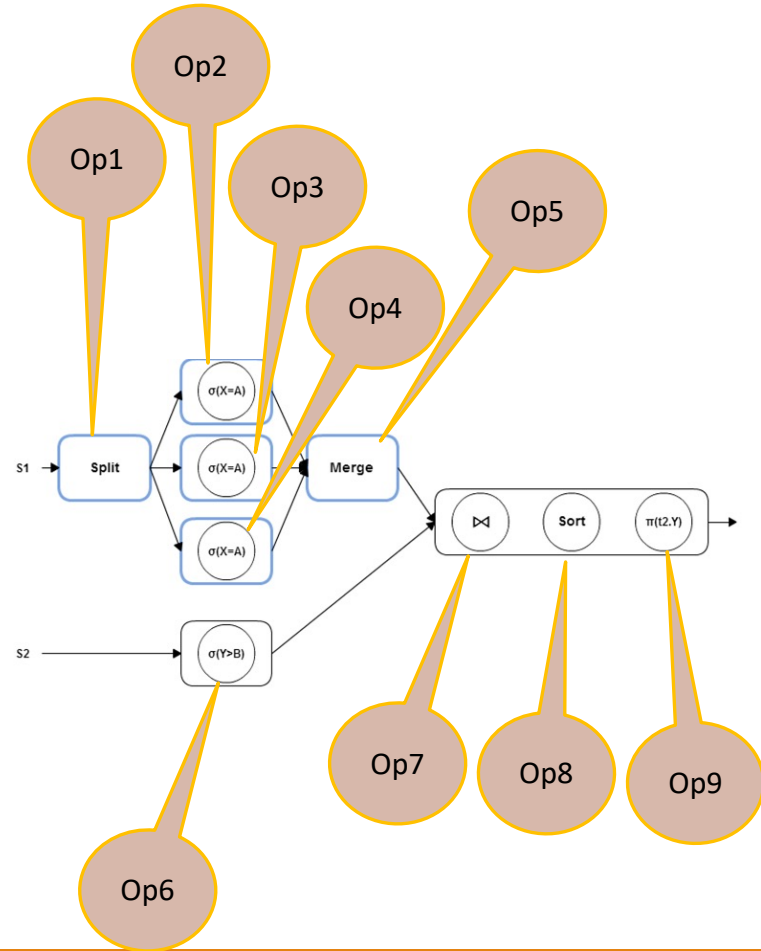
Niveau physique



Stratégie de réplication

- Réplication incrémentale
 - Ajout / suppression d'une réplique à la fois par reconfiguration
 - peut générer de nombreuses reconfigurations
- Réplication multiple
 - Ajout / suppression de plusieurs répliques à la fois par reconfiguration
 - peut générer de fortes variations de ressources

Stratégie d'allocation



Moteur de requête distribué

Machine M_1

Cœur C_{11} Op₁

Cœur C_{12} Op₂

Cœur C_{13} Op₄

Machine M_2

Cœur C_{21} Op₃

Cœur C_{22}

Machine M_k

Cœur C_{k1}

Cœur C_{k2}

Machine M_n

Cœur C_{n1} Op₉

Cœur C_{n2}

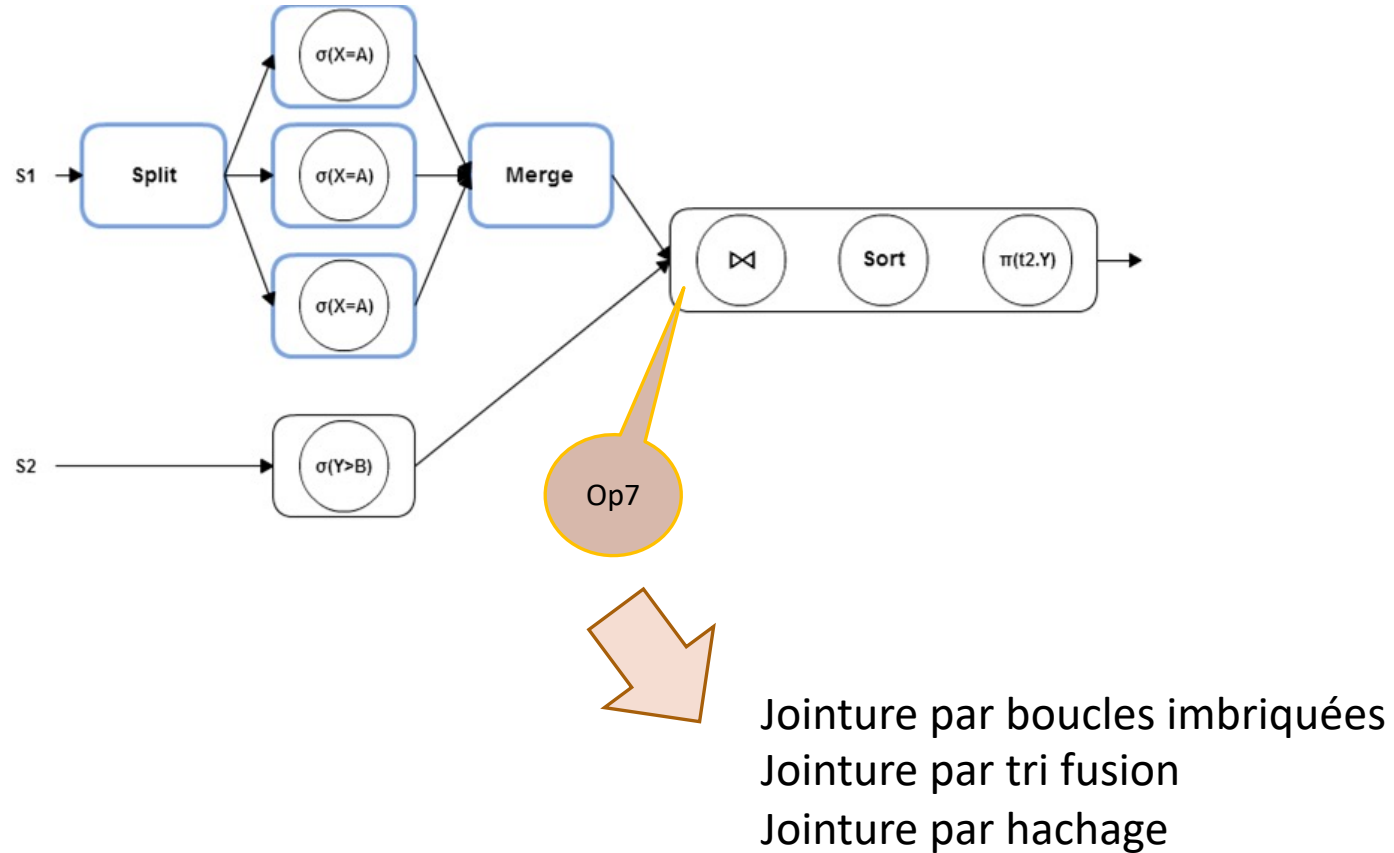
Cœur C_{n3}

Stratégies d'allocation

Différentes stratégies :

- Orienté répartition
 - *Round robin* des affectations des threads sur les unités de traitements
 - + équilibre la charge (si les temps de traitement des tuples considérés comme identiques)
 - Pas de considération du réseau
- Orienté échange réseau
 - Tient compte de la position des opérateurs dans la topologie pour choisir leur affectation
 - + réduit le trafic réseau
 - concentre la charge sur certaines unités de traitement
- Orienté ressources
 - Tient compte des ressources (CPU, RAM) disponibles sur une unité de traitement
 - + occupation maximale d'un sous-ensemble minimal de ressources (algo sac à dos)
 - Faible robustesse en cas de variation du flux

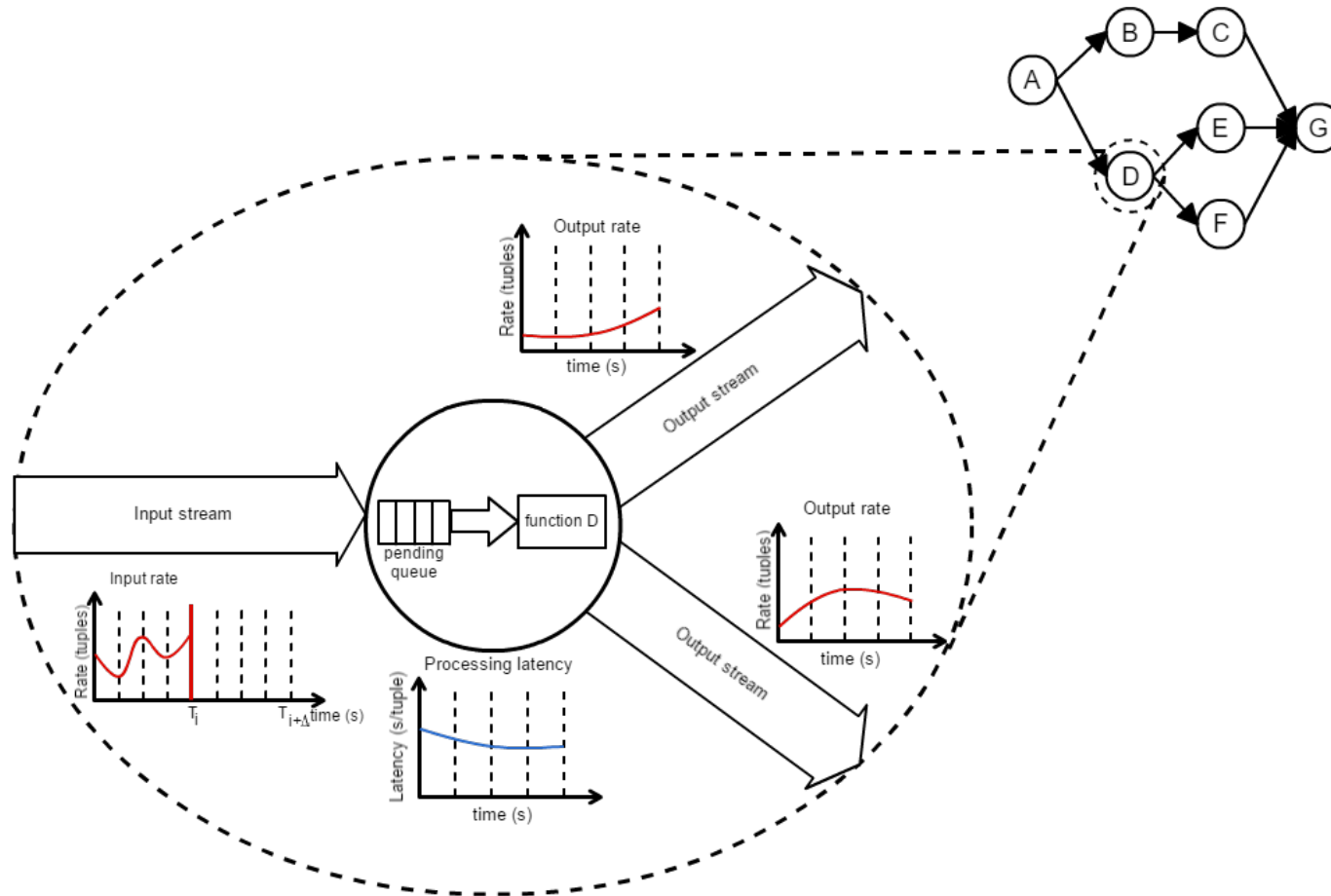
Choix de l'implémentation



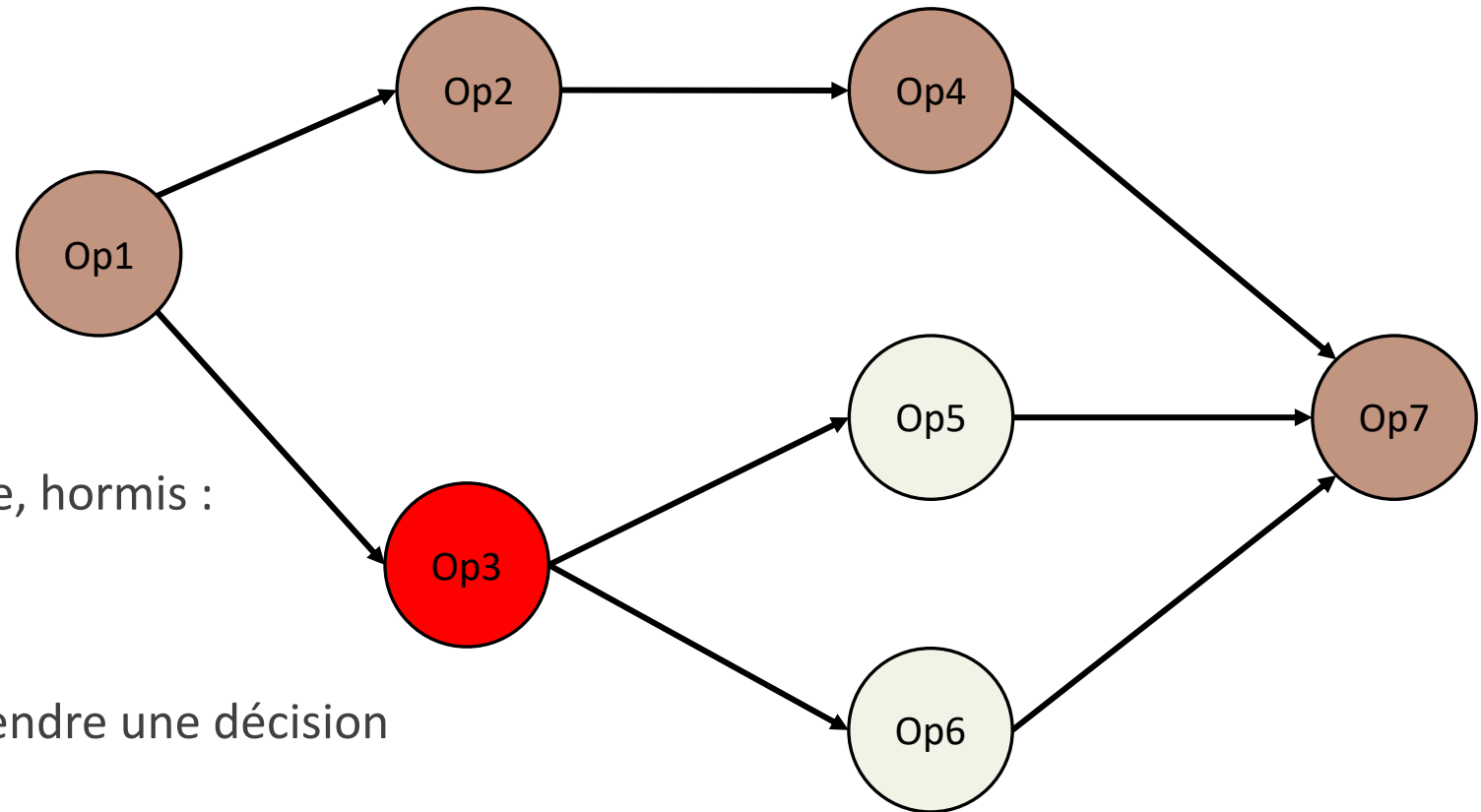
Focus sur le degré de parallélisme

- ▶ Changer le degré de parallélisme d'un opérateur.
 - ▶ Quand ?
 - ▶ À la détection de la congestion de l'opérateur (approche curative)
 - ▶ À la détection d'un risque de congestion de l'opérateur (approche préventive)
 - ▶ Comment ?
 - ▶ En répliquant un opérateur (scale out) / supprimant une réplique (scale in)

Monitorer un opérateur



Question !



Tous les opérateurs ont une charge normale, hormis :

- Op3 qui est en surcharge
- Op5 et Op6 qui est en sous utilisation

Quelle reconfiguration faire ? Comment prendre une décision

Coût de la reconfiguration

Reconfigurer le système, c'est :

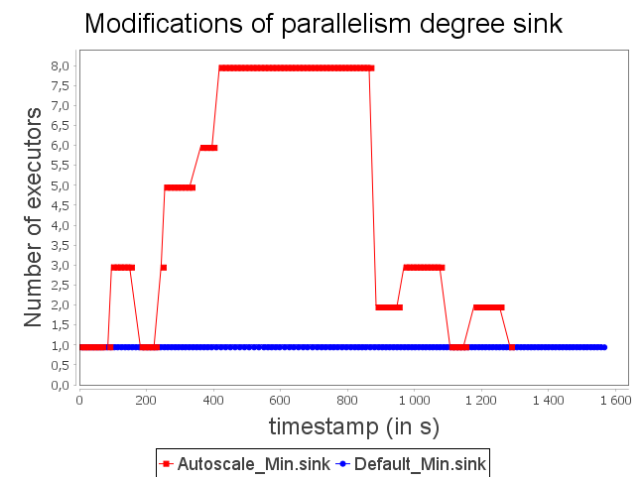
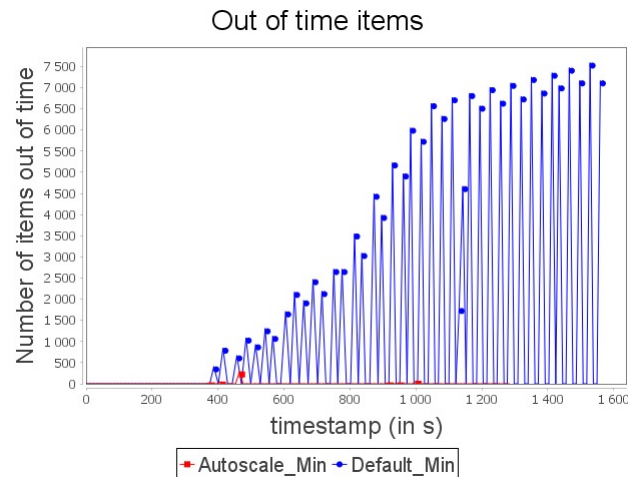
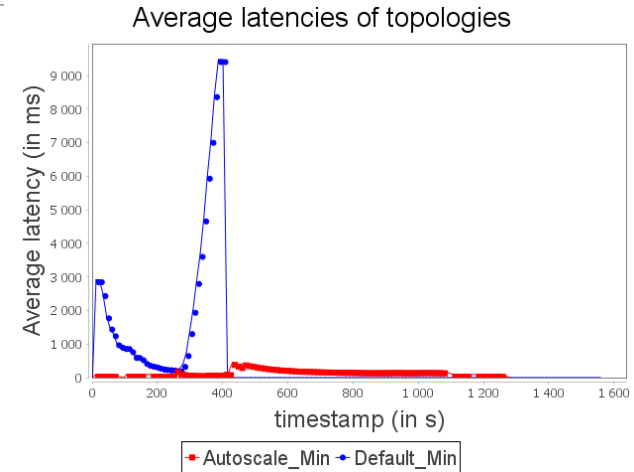
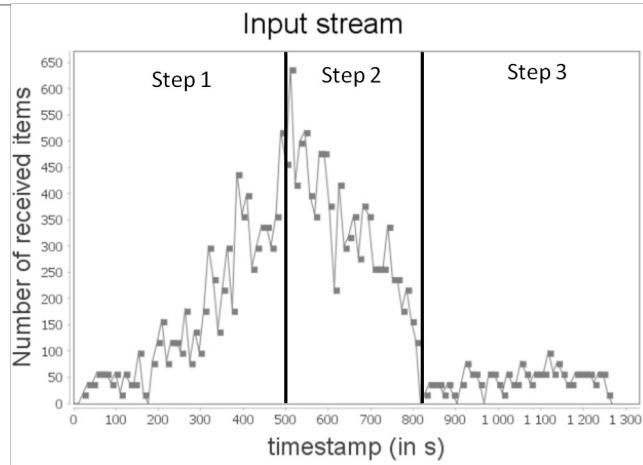
- Exploiter les informations sur l'état du système
- Faire le calcul de la meilleur reconfiguration
 - En fonction des caractéristiques des flux en entrée
- Appliquer la nouvelle reconfiguration



Sur-réactivité du système pour déclencher une reconfiguration

Quelques résultats sur l'élasticité

Configuration Min



Annexe

Apache STORM

DSMS distribué et open-source

Projet Apache débuté en février 2014 (basé sur des travaux initiés en 2011)

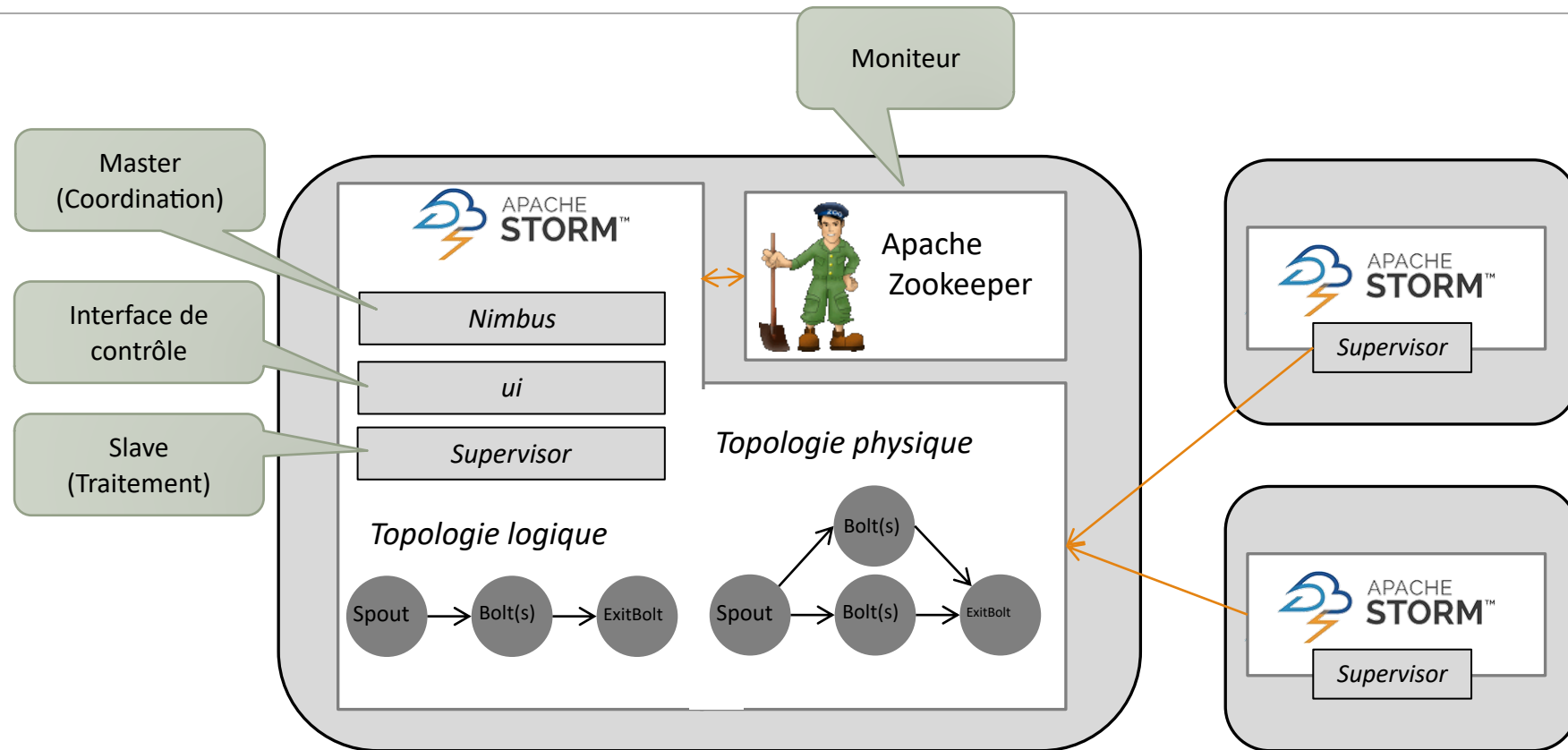
Dernière *released* septembre 2022

Principe basé sur l'exécution d'une topologie

- Graphe orienté acyclique composé
 - de nœuds (représentants les opérateurs logiques)
 - d'arcs (représentants les transferts de flux)

API multilangages (JAVA, SCALA, Python, Clojure...)

Architecture



Les nœuds logiques de Storm

Storm fournit deux primitives pour le traitement des flux :

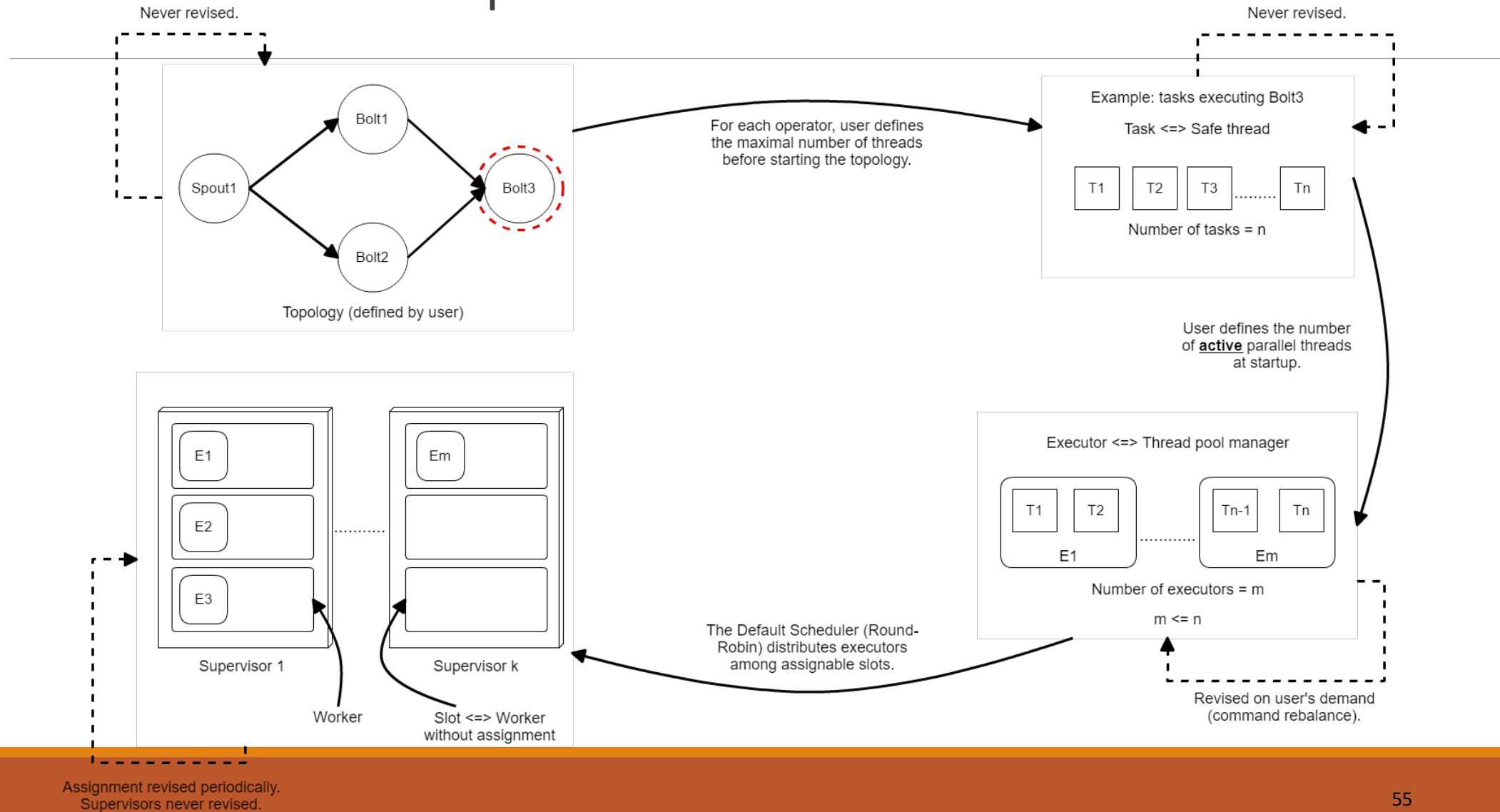
- Le nœud **Spout** : se connecte à une source de flux (ex : API Twitter) et diffuse le flux à un certain nombre d'autres nœuds.
- Le nœud **Bolt** : consomme un nombre variable de flux, effectue un traitement (application de fonctions, filtrage...) et renvoie un résultat (qui peut être un flux).

Les nœuds physiques de Storm

Deux types de nœuds physiques :

- Le Master node : exécute un processus Nimbus chargé de la distribution du code dans le cluster, de l'assignation des tâches et du monitoring.
- Le Worker node : exécute un processus Supervisor chargé de décider de l'ordre d'exécution des tâches affectées aux workers.

Les concepts dans STORM



Traitement de requêtes

Optimisation logique

- Définition de la topologie

Choix du degré de parallélisme des opérateurs

- Définition du nombre d'*executors* associés à chaque opérateur (nombre de tâches)
- Définition du nombre de *worker* associé à la topologie (nombre d'unités de traitement)

Choix de l'allocation d'une tâche sur une unité de traitement

- Définition d'une stratégie d'allocation dans le *Scheduler*

Extensions

T-Storm

- Orienté trafic

R-Storm

- Orienté ressource
 - (implémenté dans version Apache STORM 1.X)

CE-Storm

STELA

- Orienté traitement de la congestion d'opérateur

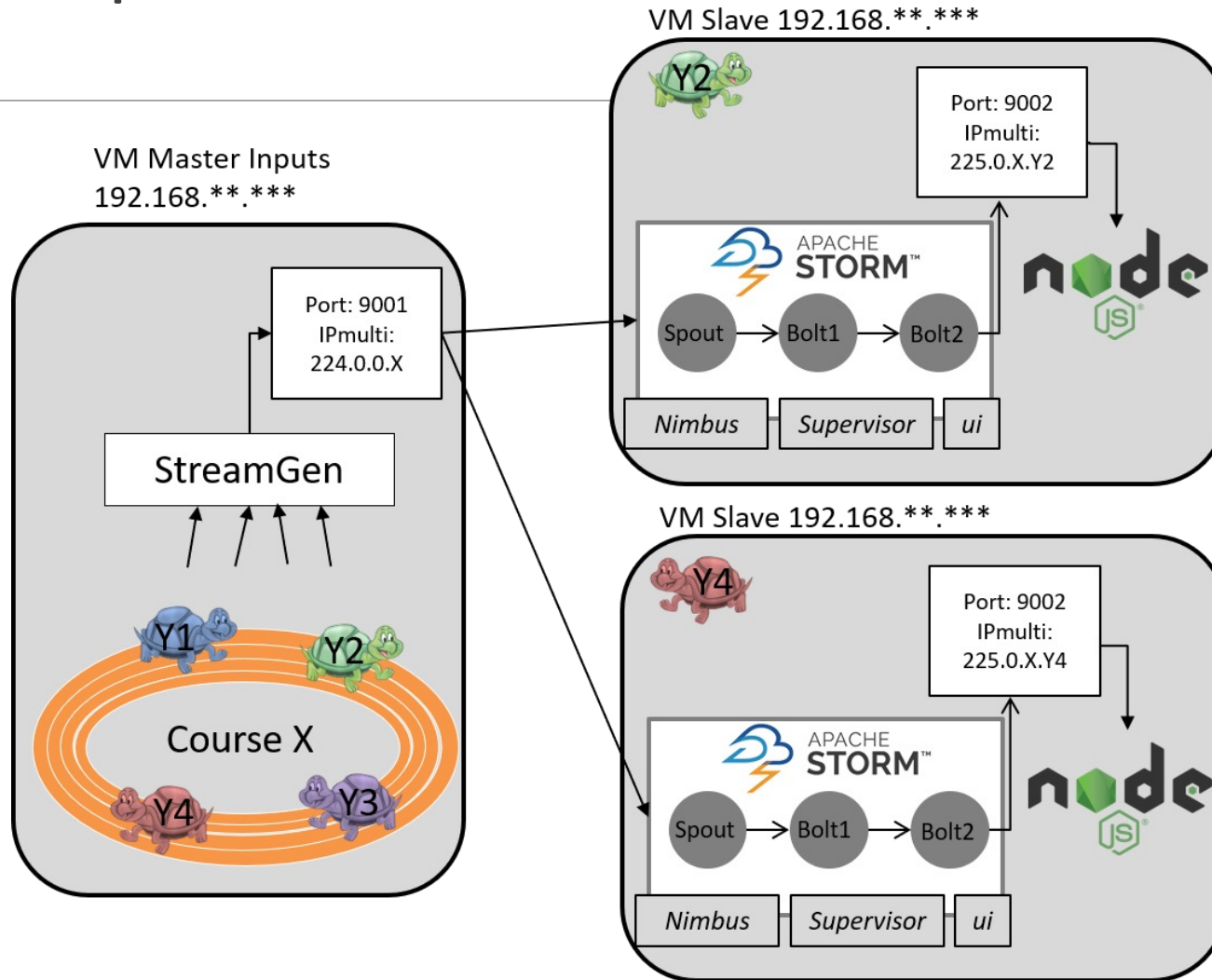
...

API Storm

Avec Apache STORM, il est possible d'implémenter :

- Des opérateurs
 - 'Stateless'
 - Avec ou sans fenêtre
 - 'Stateful'
 - Avec ou sans fenêtre
- Des fenêtres basées :
 - Sur le nombre de tuples
 - Sur une durée

A propos du TP



En TP

Flux constant en entrée, fréquence régulière

```
ubuntu@masteroutput: ~  
{ "id":2,"top":1368,"position":205,"nbDevant":0,"nbDerriere":9,"total":10},  
{ "id":3,"top":1368,"position":180,"nbDevant":3,"nbDerriere":6,"total":10},  
{ "id":4,"top":1368,"position":186,"nbDevant":2,"nbDerriere":7,"total":10},  
{ "id":5,"top":1368,"position":200,"nbDevant":1,"nbDerriere":8,"total":10},  
{ "id":6,"top":1368,"position":171,"nbDevant":5,"nbDerriere":4,"total":10},  
{ "id":7,"top":1368,"position":165,"nbDevant":6,"nbDerriere":3,"total":10},  
{ "id":8,"top":1368,"position":176,"nbDevant":4,"nbDerriere":5,"total":10},  
{ "id":9,"top":1368,"position":151,"nbDevant":9,"nbDerriere":0,"total":10}  
] }  
  
[2017-11-16 16:39:37]{ tortoises:[  
{ "id":0,"top":1369,"position":163,"nbDevant":7,"nbDerriere":1,"total":10},  
{ "id":1,"top":1369,"position":163,"nbDevant":7,"nbDerriere":1,"total":10},  
{ "id":2,"top":1369,"position":205,"nbDevant":0,"nbDerriere":9,"total":10},  
{ "id":3,"top":1369,"position":181,"nbDevant":3,"nbDerriere":6,"total":10},  
{ "id":4,"top":1369,"position":186,"nbDevant":2,"nbDerriere":7,"total":10},  
{ "id":5,"top":1369,"position":200,"nbDevant":1,"nbDerriere":8,"total":10},  
{ "id":6,"top":1369,"position":171,"nbDevant":5,"nbDerriere":4,"total":10},  
{ "id":7,"top":1369,"position":166,"nbDevant":6,"nbDerriere":3,"total":10},  
{ "id":8,"top":1369,"position":176,"nbDevant":4,"nbDerriere":5,"total":10},  
{ "id":9,"top":1369,"position":151,"nbDevant":9,"nbDerriere":0,"total":10}  
] }
```

exit
511.00 ms

default: 20: 50%

myTortoise
NaN ms

default: 20: 50%

masterStream
NaN ms

```
ubuntu@masteroutput: ~  
:8,"nbDerriere":1,"total":10}  
  
[2017-11-16 17:09:49>{"id":1,"top":635,"nom":"Lumineau","position":37,"nbDevant":  
:8,"nbDerriere":1,"total":10}  
  
[2017-11-16 17:09:51>{"id":1,"top":636,"nom":"Lumineau","position":37,"nbDevant":  
:8,"nbDerriere":1,"total":10}  
  
[2017-11-16 17:09:53>{"id":1,"top":637,"nom":"Lumineau","position":37,"nbDevant":  
:8,"nbDerriere":1,"total":10}  
  
[2017-11-16 17:09:55>{"id":1,"top":638,"nom":"Lumineau","position":38,"nbDevant":  
:8,"nbDerriere":1,"total":10}  
  
[2017-11-16 17:09:57>{"id":1,"top":639,"nom":"Lumineau","position":39,"nbDevant":  
:8,"nbDerriere":1,"total":10}  
  
[2017-11-16 17:09:59>{"id":1,"top":640,"nom":"Lumineau","position":40,"nbDevant":  
:8,"nbDerriere":1,"total":10}  
  
[2017-11-16 17:10:01>{"id":1,"top":641,"nom":"Lumineau","position":41,"nbDevant":  
:8,"nbDerriere":1,"total":10}
```

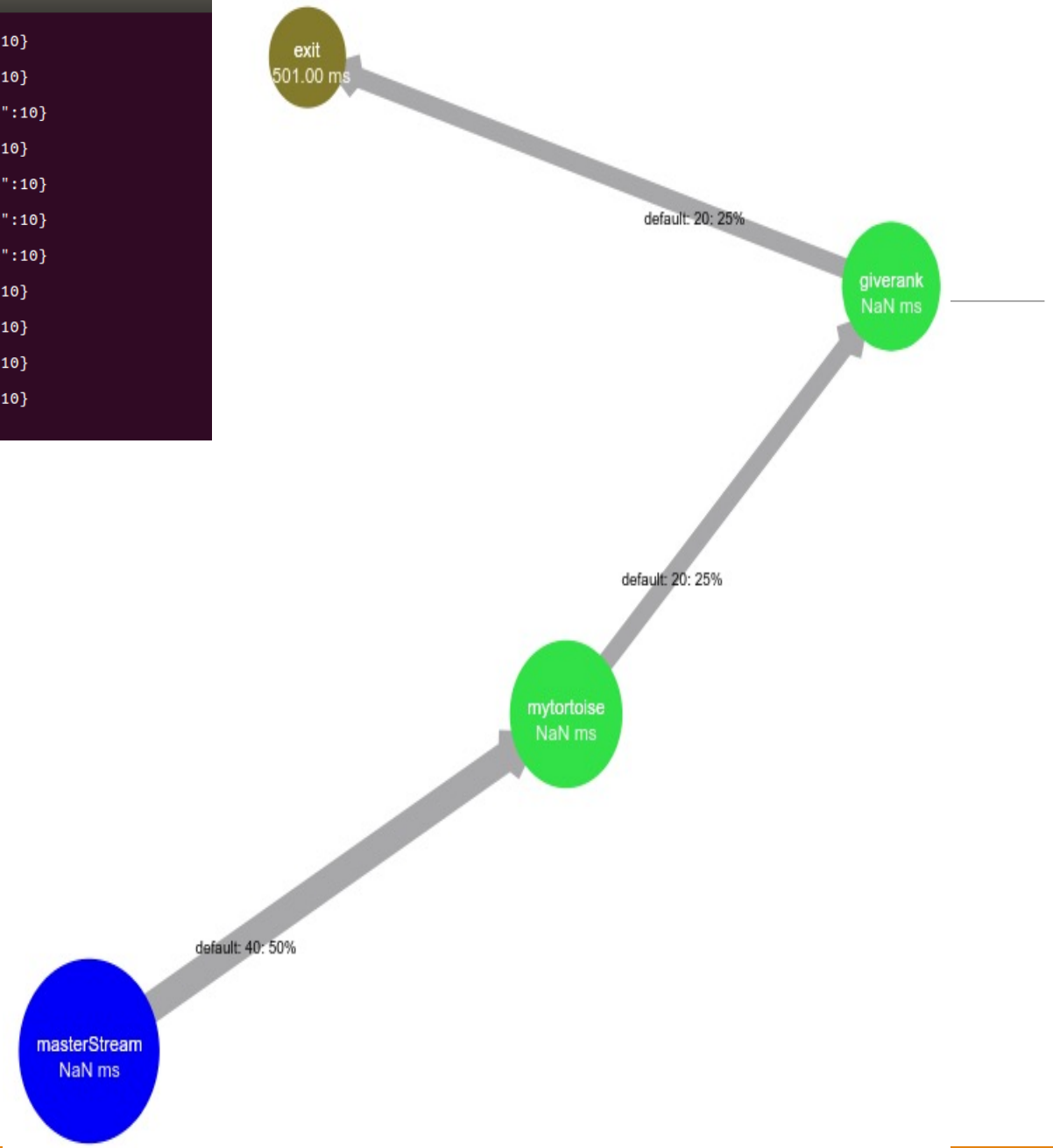


```
ubuntu@masteroutput: ~  
{  
  "id":2,"top":1368,"position":205,"nbDevant":0,"nbDerriere":9,"total":10},  
  {"id":3,"top":1368,"position":180,"nbDevant":3,"nbDerriere":6,"total":10},  
  {"id":4,"top":1368,"position":186,"nbDevant":2,"nbDerriere":7,"total":10},  
  {"id":5,"top":1368,"position":200,"nbDevant":1,"nbDerriere":8,"total":10},  
  {"id":6,"top":1368,"position":171,"nbDevant":5,"nbDerriere":4,"total":10},  
  {"id":7,"top":1368,"position":165,"nbDevant":6,"nbDerriere":3,"total":10},  
  {"id":8,"top":1368,"position":176,"nbDevant":4,"nbDerriere":5,"total":10},  
  {"id":9,"top":1368,"position":151,"nbDevant":9,"nbDerriere":0,"total":10}  
}  
  
[2017-11-16 16:39:37]{ tortoises:  
  {"id":0,"top":1369,"position":163,"nbDevant":7,"nbDerriere":1,"total":10},  
  {"id":1,"top":1369,"position":163,"nbDevant":7,"nbDerriere":1,"total":10},  
  {"id":2,"top":1369,"position":205,"nbDevant":0,"nbDerriere":9,"total":10},  
  {"id":3,"top":1369,"position":181,"nbDevant":3,"nbDerriere":6,"total":10},  
  {"id":4,"top":1369,"position":186,"nbDevant":2,"nbDerriere":7,"total":10},  
  {"id":5,"top":1369,"position":200,"nbDevant":1,"nbDerriere":8,"total":10},  
  {"id":6,"top":1369,"position":171,"nbDevant":5,"nbDerriere":4,"total":10},  
  {"id":7,"top":1369,"position":166,"nbDevant":6,"nbDerriere":3,"total":10},  
  {"id":8,"top":1369,"position":176,"nbDevant":4,"nbDerriere":5,"total":10},  
  {"id":9,"top":1369,"position":151,"nbDevant":9,"nbDerriere":0,"total":10}  
}
```

```
ubuntu@masteroutput: ~
[2017-11-17 08:08:03]{ "id":1,"top":329,"nom":"Lumineau","rang":"9","nbTotal":10}
[2017-11-17 08:08:05]{ "id":1,"top":330,"nom":"Lumineau","rang":"9","nbTotal":10}
[2017-11-17 08:08:07]{ "id":1,"top":331,"nom":"Lumineau","rang":"9ex","nbTotal":10}
[2017-11-17 08:08:09]{ "id":1,"top":332,"nom":"Lumineau","rang":"9","nbTotal":10}
[2017-11-17 08:08:11]{ "id":1,"top":333,"nom":"Lumineau","rang":"9ex","nbTotal":10}
[2017-11-17 08:08:13]{ "id":1,"top":334,"nom":"Lumineau","rang":"9ex","nbTotal":10}
[2017-11-17 08:08:15]{ "id":1,"top":335,"nom":"Lumineau","rang":"9ex","nbTotal":10}
[2017-11-17 08:08:17]{ "id":1,"top":336,"nom":"Lumineau","rang":"9","nbTotal":10}
[2017-11-17 08:08:19]{ "id":1,"top":337,"nom":"Lumineau","rang":"9","nbTotal":10}
[2017-11-17 08:08:21]{ "id":1,"top":338,"nom":"Lumineau","rang":"9","nbTotal":10}
[2017-11-17 08:08:23]{ "id":1,"top":339,"nom":"Lumineau","rang":"9","nbTotal":10}
```

```
ubuntu@masteroutput: ~
:8,"nbDerriere":1,"total":10}
[2017-11-16 17:09:49]{ "id":1,"top":635,"nom":"Lumineau","position":37,"nbDevant":
:8,"nbDerriere":1,"total":10}
[2017-11-16 17:09:51]{ "id":1,"top":636,"nom":"Lumineau","position":37,"nbDevant":
:8,"nbDerriere":1,"total":10}
[2017-11-16 17:09:53]{ "id":1,"top":637,"nom":"Lumineau","position":37,"nbDevant":
:8,"nbDerriere":1,"total":10}
[2017-11-16 17:09:55]{ "id":1,"top":638,"nom":"Lumineau","position":38,"nbDevant":
:8,"nbDerriere":1,"total":10}
[2017-11-16 17:09:57]{ "id":1,"top":639,"nom":"Lumineau","position":39,"nbDevant":
:8,"nbDerriere":1,"total":10}
[2017-11-16 17:09:59]{ "id":1,"top":640,"nom":"Lumineau","position":40,"nbDevant":
:8,"nbDerriere":1,"total":10}
[2017-11-16 17:10:01]{ "id":1,"top":641,"nom":"Lumineau","position":41,"nbDevant":
:8,"nbDerriere":1,"total":10}
```

```
ubuntu@masteroutput: ~
{"id":2,"top":1368,"position":285,"nbDevant":0,"nbDerriere":9,"total":10},
{"id":3,"top":1368,"position":180,"nbDevant":3,"nbDerriere":6,"total":10},
{"id":4,"top":1368,"position":186,"nbDevant":2,"nbDerriere":7,"total":10},
{"id":5,"top":1368,"position":200,"nbDevant":1,"nbDerriere":8,"total":10},
{"id":6,"top":1368,"position":171,"nbDevant":5,"nbDerriere":4,"total":10},
{"id":7,"top":1368,"position":165,"nbDevant":6,"nbDerriere":3,"total":10},
{"id":8,"top":1368,"position":176,"nbDevant":4,"nbDerriere":5,"total":10},
{"id":9,"top":1368,"position":151,"nbDevant":9,"nbDerriere":0,"total":10}
] }
[2017-11-16 16:39:37]{ tortoises:[
{"id":0,"top":1369,"position":163,"nbDevant":7,"nbDerriere":1,"total":10},
{"id":1,"top":1369,"position":163,"nbDevant":7,"nbDerriere":1,"total":10},
{"id":2,"top":1369,"position":285,"nbDevant":0,"nbDerriere":9,"total":10},
{"id":3,"top":1369,"position":181,"nbDevant":3,"nbDerriere":6,"total":10},
{"id":4,"top":1369,"position":186,"nbDevant":2,"nbDerriere":7,"total":10},
{"id":5,"top":1369,"position":200,"nbDevant":1,"nbDerriere":8,"total":10},
{"id":6,"top":1369,"position":171,"nbDevant":5,"nbDerriere":4,"total":10},
{"id":7,"top":1369,"position":166,"nbDevant":6,"nbDerriere":3,"total":10},
{"id":8,"top":1369,"position":176,"nbDevant":4,"nbDerriere":5,"total":10},
{"id":9,"top":1369,"position":151,"nbDevant":9,"nbDerriere":0,"total":10}
] }
```




```

[2017-11-17 08:30:36]{ "id":1,"nom":"Lumineau","tops":"996-1005","vitesse":0.55}
[2017-11-17 08:30:38]{ "id":1,"nom":"Lumineau","tops":"997-1006","vitesse":0.55}
[2017-11-17 08:30:40]{ "id":1,"nom":"Lumineau","tops":"998-1007","vitesse":0.55}
[2017-11-17 08:30:42]{ "id":1,"nom":"Lumineau","tops":"999-1008","vitesse":0.44}
[2017-11-17 08:30:44]{ "id":1,"nom":"Lumineau","tops":"1000-1009","vitesse":0.44}
[2017-11-17 08:30:46]{ "id":1,"nom":"Lumineau","tops":"1001-1010","vitesse":0.33}
[2017-11-17 08:30:48]{ "id":1,"nom":"Lumineau","tops":"1002-1011","vitesse":0.44}
[2017-11-17 08:30:50]{ "id":1,"nom":"Lumineau","tops":"1003-1012","vitesse":0.44}
[2017-11-17 08:30:52]{ "id":1,"nom":"Lumineau","tops":"1004-1013","vitesse":0.44}
[2017-11-17 08:30:54]{ "id":1,"nom":"Lumineau","tops":"1005-1014","vitesse":0.44}
[2017-11-17 08:30:56]{ "id":1,"nom":"Lumineau","tops":"1006-1015","vitesse":0.55}

```

```

ubuntu@masteroutput: ~
:8,"nbDerriere":1,"total":10}

[2017-11-16 17:09:49]{ "id":1,"top":635,"nom":"Lumineau","position":37,"nbDevant"
:8,"nbDerriere":1,"total":10}

[2017-11-16 17:09:51]{ "id":1,"top":636,"nom":"Lumineau","position":37,"nbDevant"
:8,"nbDerriere":1,"total":10}

[2017-11-16 17:09:53]{ "id":1,"top":637,"nom":"Lumineau","position":37,"nbDevant"
:8,"nbDerriere":1,"total":10}

[2017-11-16 17:09:55]{ "id":1,"top":638,"nom":"Lumineau","position":38,"nbDevant"
:8,"nbDerriere":1,"total":10}

[2017-11-16 17:09:57]{ "id":1,"top":639,"nom":"Lumineau","position":39,"nbDevant"
:8,"nbDerriere":1,"total":10}

[2017-11-16 17:09:59]{ "id":1,"top":640,"nom":"Lumineau","position":40,"nbDevant"
:8,"nbDerriere":1,"total":10}

[2017-11-16 17:10:01]{ "id":1,"top":641,"nom":"Lumineau","position":41,"nbDevant"
:8,"nbDerriere":1,"total":10}

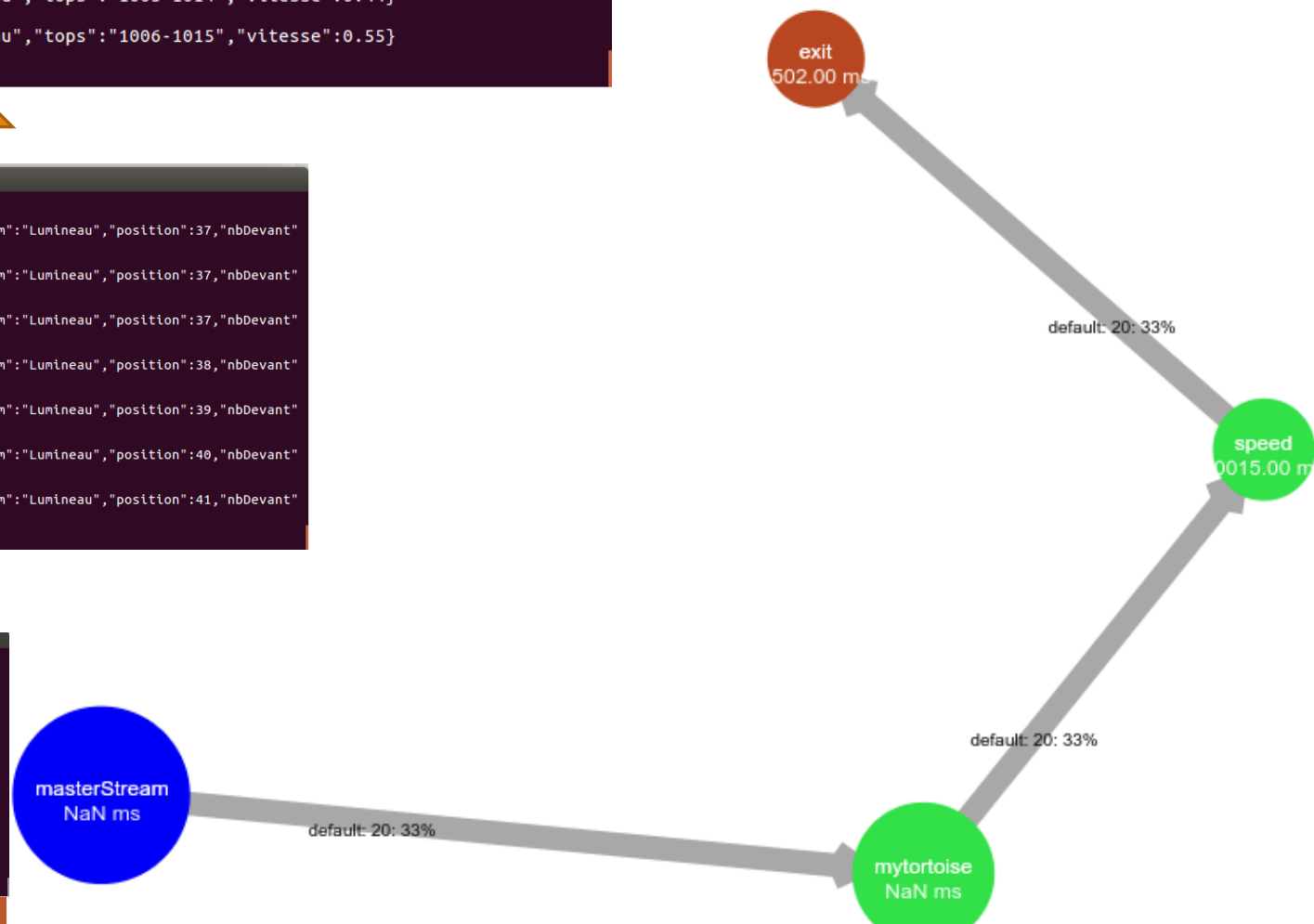
```

```

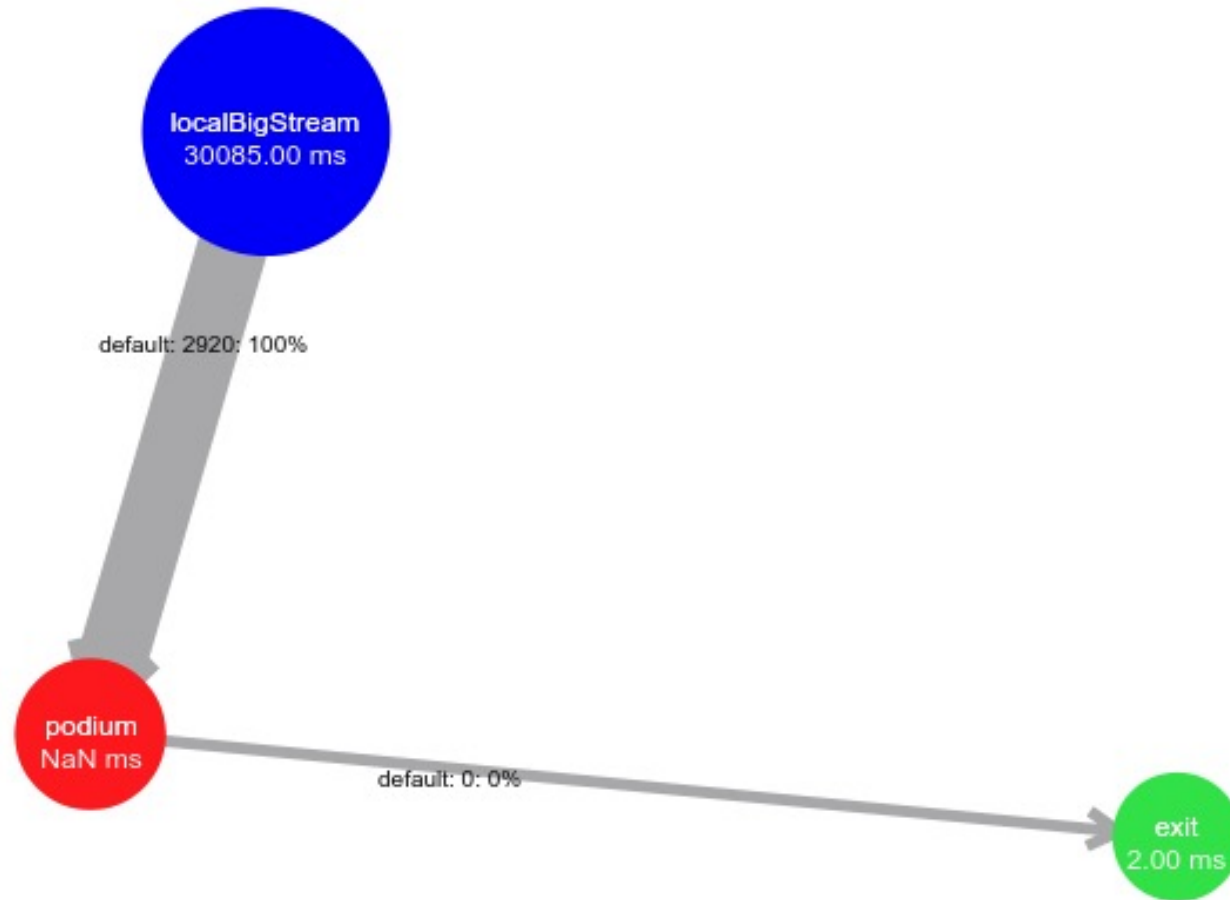
ubuntu@masteroutput: ~
{"id":2,"top":1368,"position":205,"nbDevant":0,"nbDerriere":9,"total":10},
{"id":3,"top":1368,"position":180,"nbDevant":3,"nbDerriere":6,"total":10},
{"id":4,"top":1368,"position":186,"nbDevant":2,"nbDerriere":7,"total":10},
{"id":5,"top":1368,"position":206,"nbDevant":1,"nbDerriere":8,"total":10},
{"id":6,"top":1368,"position":171,"nbDevant":5,"nbDerriere":4,"total":10},
{"id":7,"top":1368,"position":165,"nbDevant":6,"nbDerriere":3,"total":10},
{"id":8,"top":1368,"position":116,"nbDevant":4,"nbDerriere":5,"total":10},
{"id":9,"top":1368,"position":151,"nbDevant":9,"nbDerriere":0,"total":10}
}

[2017-11-16 10:39:37]{ tortoise:
{"id":0,"top":1369,"position":163,"nbDevant":7,"nbDerriere":1,"total":10},
{"id":1,"top":1369,"position":163,"nbDevant":7,"nbDerriere":1,"total":10},
{"id":2,"top":1369,"position":205,"nbDevant":9,"nbDerriere":9,"total":10},
{"id":3,"top":1369,"position":181,"nbDevant":3,"nbDerriere":6,"total":10},
{"id":4,"top":1369,"position":186,"nbDevant":2,"nbDerriere":7,"total":10},
{"id":5,"top":1369,"position":206,"nbDevant":1,"nbDerriere":8,"total":10},
{"id":6,"top":1369,"position":171,"nbDevant":5,"nbDerriere":4,"total":10},
{"id":7,"top":1369,"position":166,"nbDevant":6,"nbDerriere":3,"total":10},
{"id":8,"top":1369,"position":116,"nbDevant":4,"nbDerriere":5,"total":10},
{"id":9,"top":1369,"position":151,"nbDevant":9,"nbDerriere":0,"total":10}
}
}

```



Gestion de la congestion



Questions

Quels sont les facteurs qui peuvent influencer la capacité de traitement d'un opérateur ?

Quelle stratégie d'allocation n'est pas compatible avec le principe de libérer les ressources (scale in) ?

Pourquoi est-il risqué de considérer individuellement le changement de degré de parallélisme des opérateurs ?

Quels sont les risques d'automatiser le processus d'élasticité ?

Quels sont les facteurs influençant la résolution de la congestion du système ?