



# PROJET SR05:

## Vente aux enchères

Clément Ferry  
Yanyan Nie  
Jiayang WANG

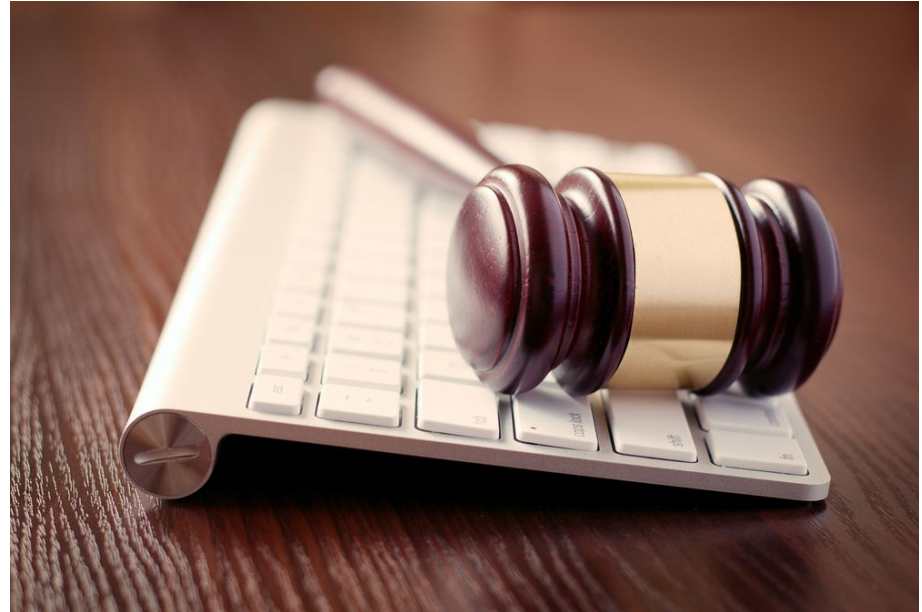


# Sommaire

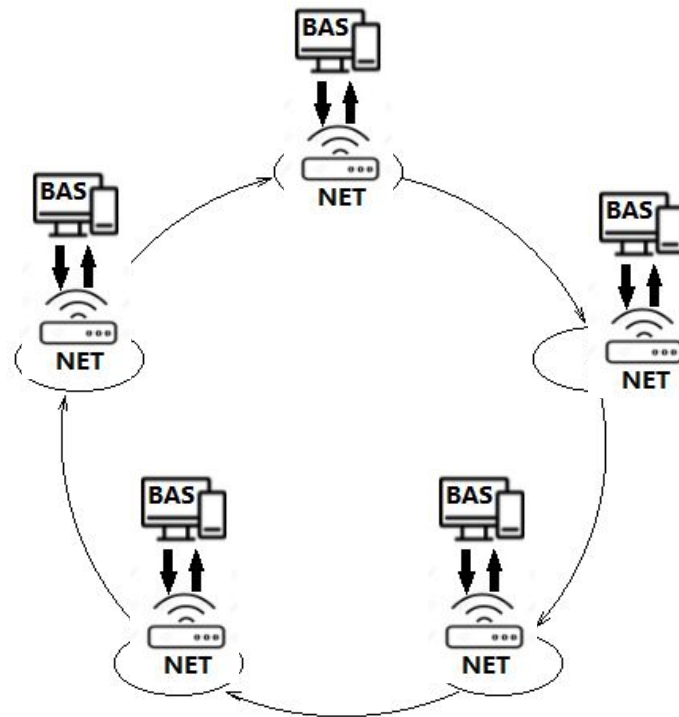
- I. Présentation du sujet
- II. Architecture
- III. Structures de données
- IV. Algorithme
- Démonstration

# I. Présentation du sujet

- Vente aux enchères
- un prix maximal dans chaque site
- but : prix le plus élevé



## II. Architecture - Ring unidirectional



### III. Structure de données

- ❑ what
- ❑ who
- ❑ where

what/who/where

`^key1~value1^key2~value2^/who/where`

EX1:

`^synchro~true^prix~61^/1/4`

`^synchro~true^prix~61^appnet~BAS^/1/4`

`^synchro~true^prix~61^appnet~BAS^/2/4`

`^synchro~true^prix~61^appnet~BAS^/3/4`

`^synchro~true^prix~61^/4/4:`

EX2:

`^prix~61^file~2;3;4^/1/1`

`^prix~65^file~3;4;1^/2/2`

`^prix~67^file~4;1^/3/3`

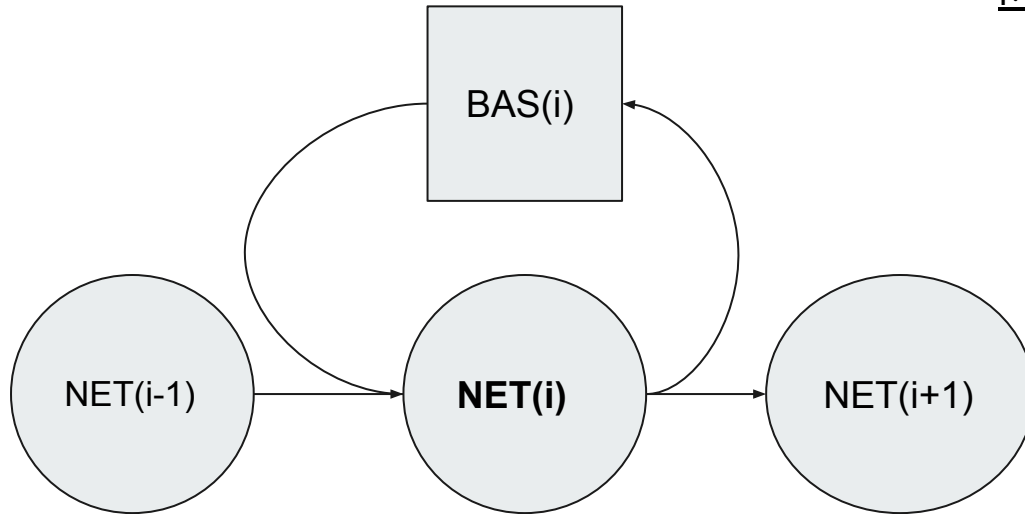


## IV. Algorithme

### Réseau: Script bash

- Liaisons FIFO
- Chaque NET reçoit les messages de son BAS et du NET précédent via le même FIFO

## IV. Algorithme de NET



$\text{^prix} \sim 61 \text{^appnet} \sim \text{BAS}^{1/4}$   $\text{^prix} \sim 61 \text{^appnet} \sim \text{BAS}^{1/4}$

### Message reçu:

Si who != where:

Si appnet == "BAS":

Si where = NAME:

Transmettre à BASE

# ie envoi sans appnet

FinSi

Sinon: #Message reçu de BAS(i)

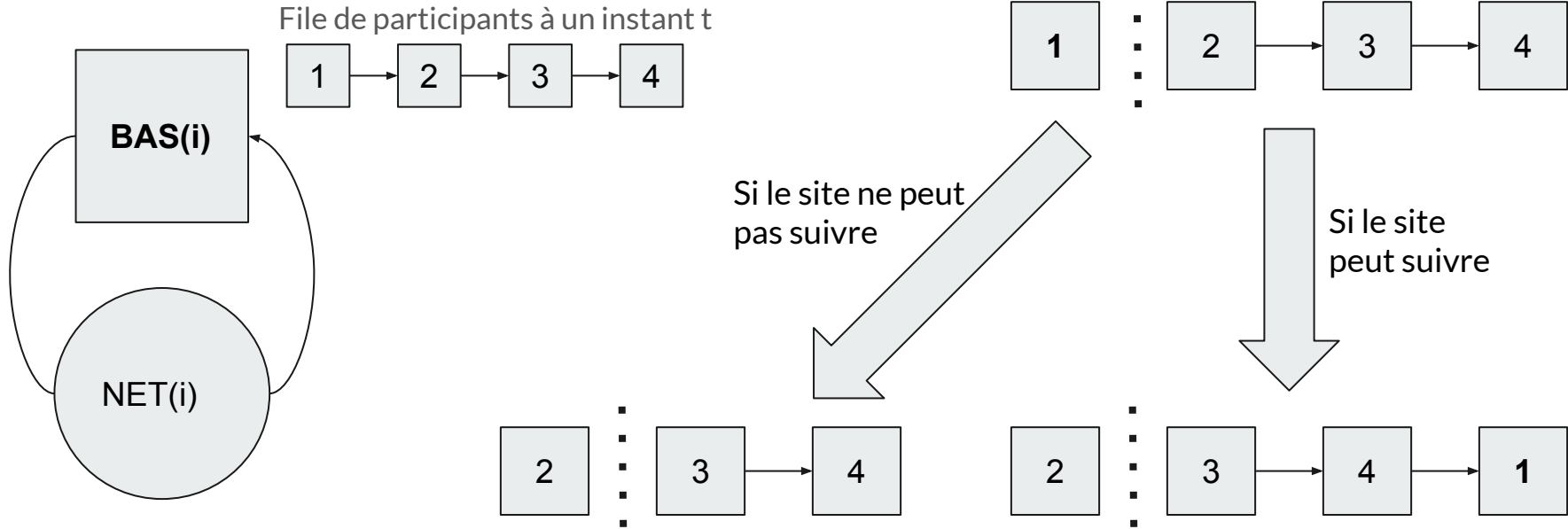
Transmission à NET

# ie envoi avec appnet

FinSi

FinSi

## IV. Algorithme de BAS



**Terminaison:** File vide (ie plus personne ne peut suivre)  
Le site qui a la main est donc le gagnant





## IV. Algorithme: Synchronisation

Lors d'un changement de prix:

- 1) Envoi du nouveau prix avec un message de type "synchronisation"
- 2) Envoi d'un message au prochain participant de la pile pour qu'il propose un prix



## IV. Algorithme: Snapshot (Algorithme de marqueur)

Démarrage: envoi d'un message de type snapshot à un BAS

Transmission de la date de demande du snapshot



# Conclusion

Possibilité de généraliser à une topologie de réseau complet facilement



# Démonstration