

# DU ISN2

## Projet de programmation

---

**Partie client-serveur**

**Vania Marangozova**

**Juin 2020**

# Architecture centralisée de votre projet

---

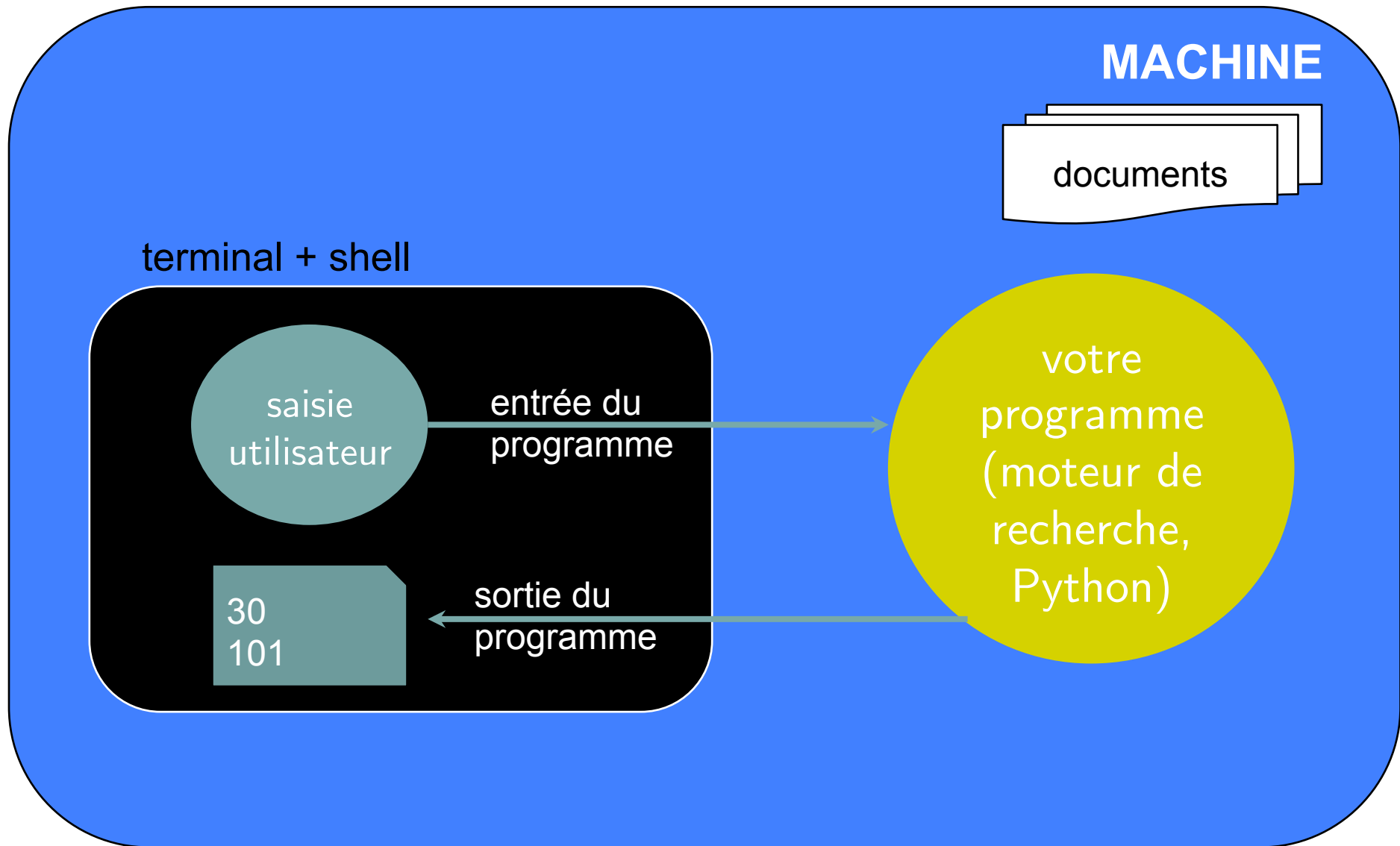
## ► Un exemple d'exécution

! Attention, la réponse est inventée et non calculée :)

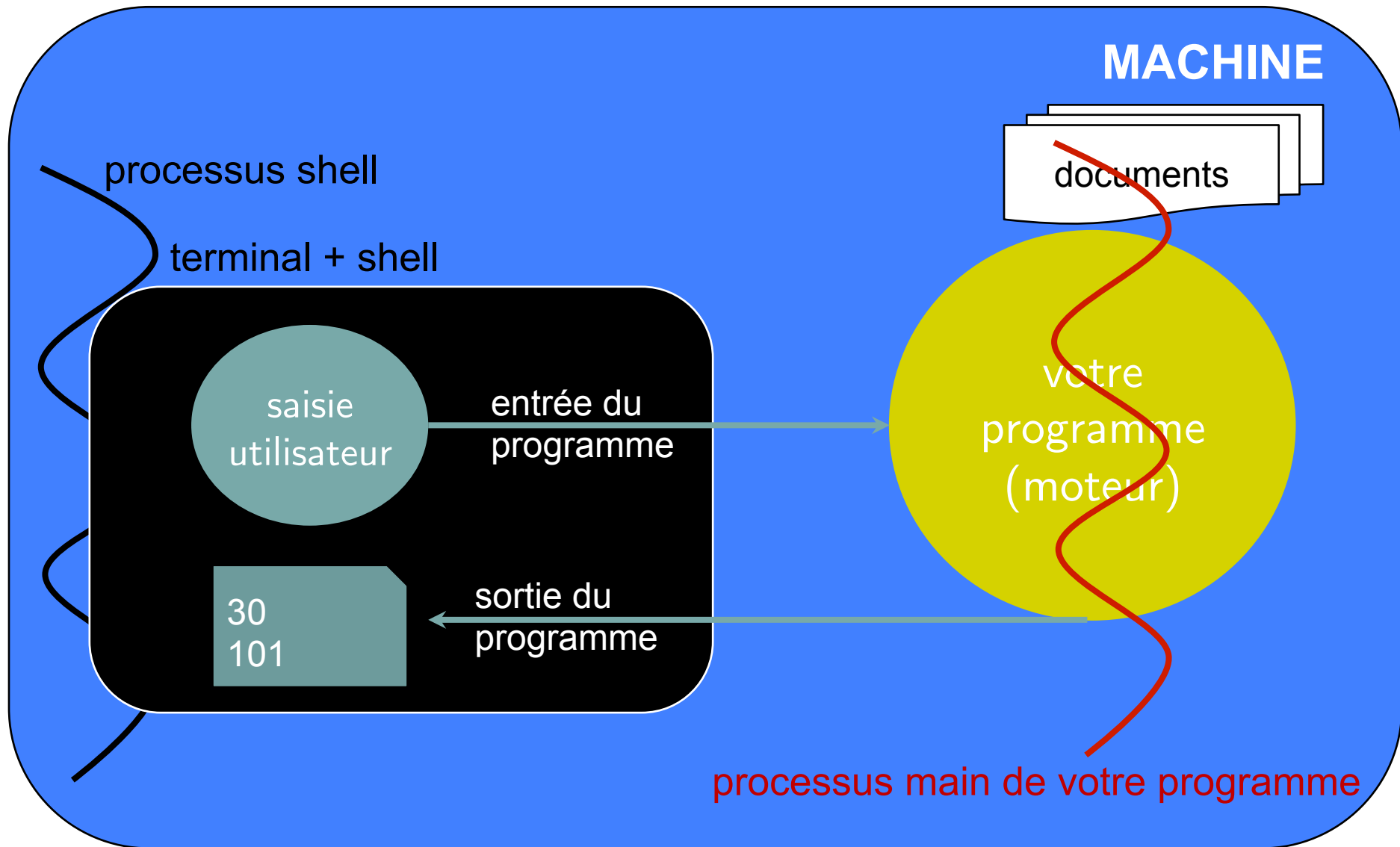
```
> ./recherche.py
Entrer le mot que vous recherchez : chat
Les documents qui contiennent "chat" sont :
30
404
1034
1
Salut!
>
```

# Architecture centralisée de votre projet

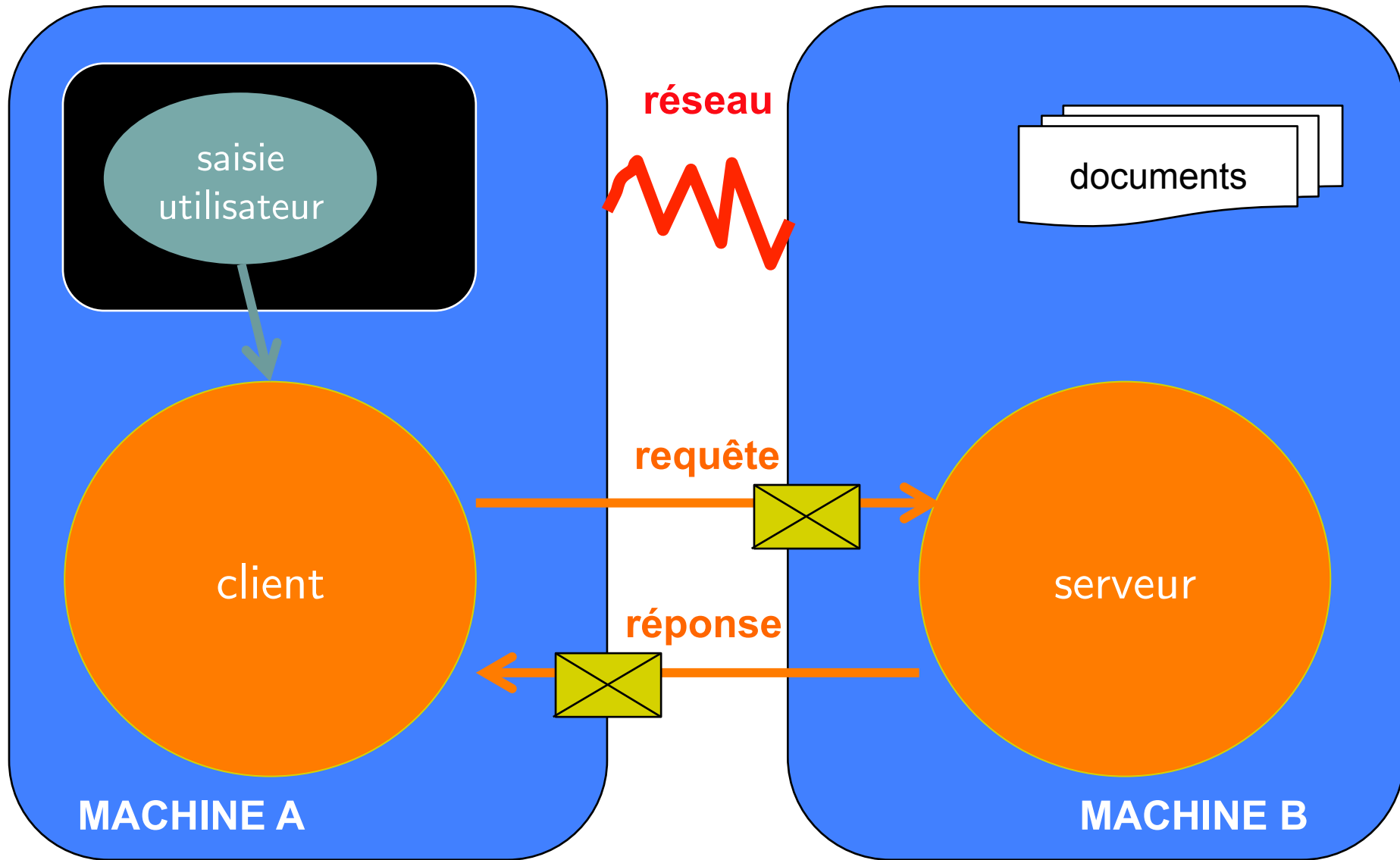
---



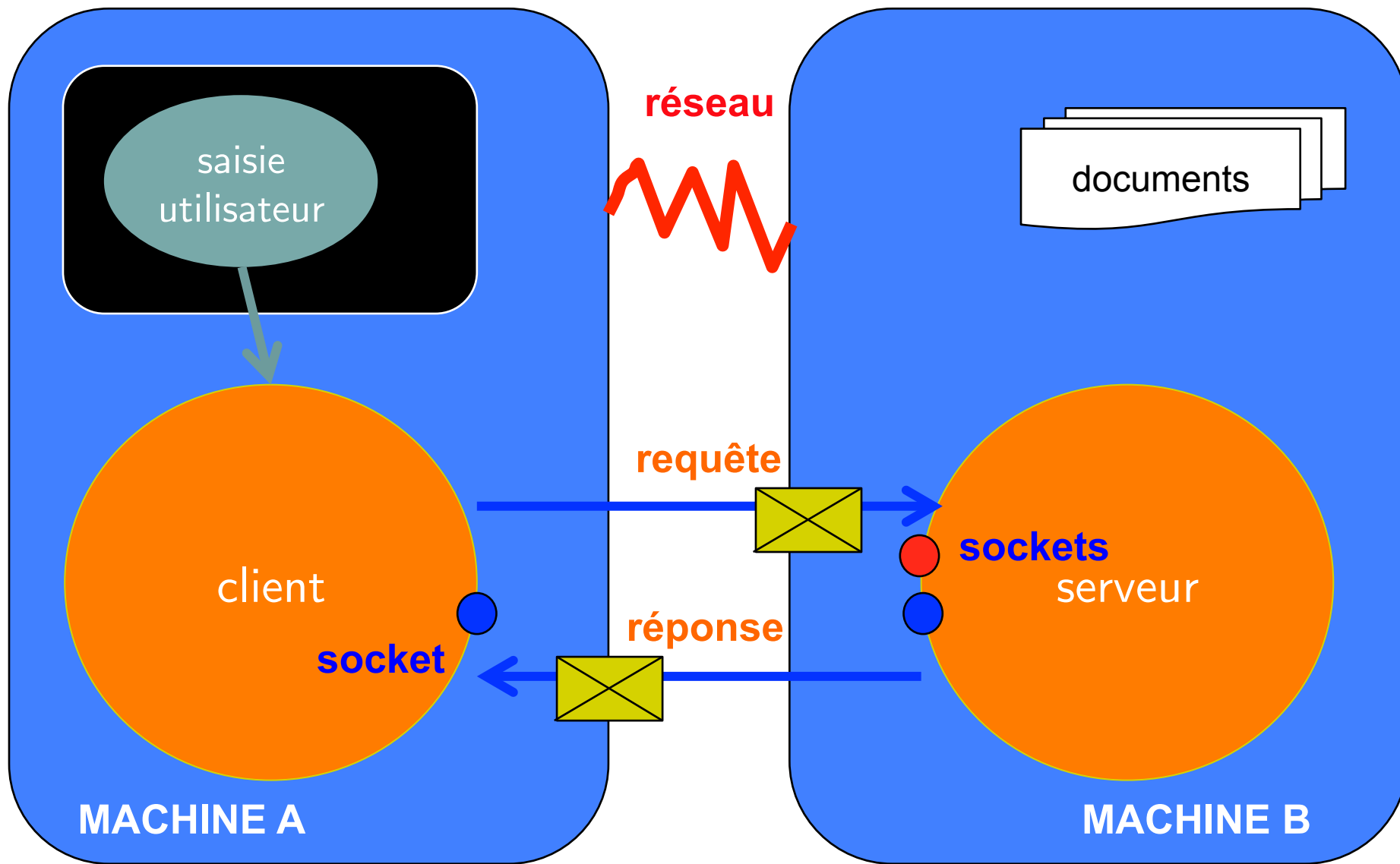
# Architecture centralisée de votre projet



# Architecture client-serveur (répartie) de votre projet



# Architecture client-serveur (répartie) de votre projet

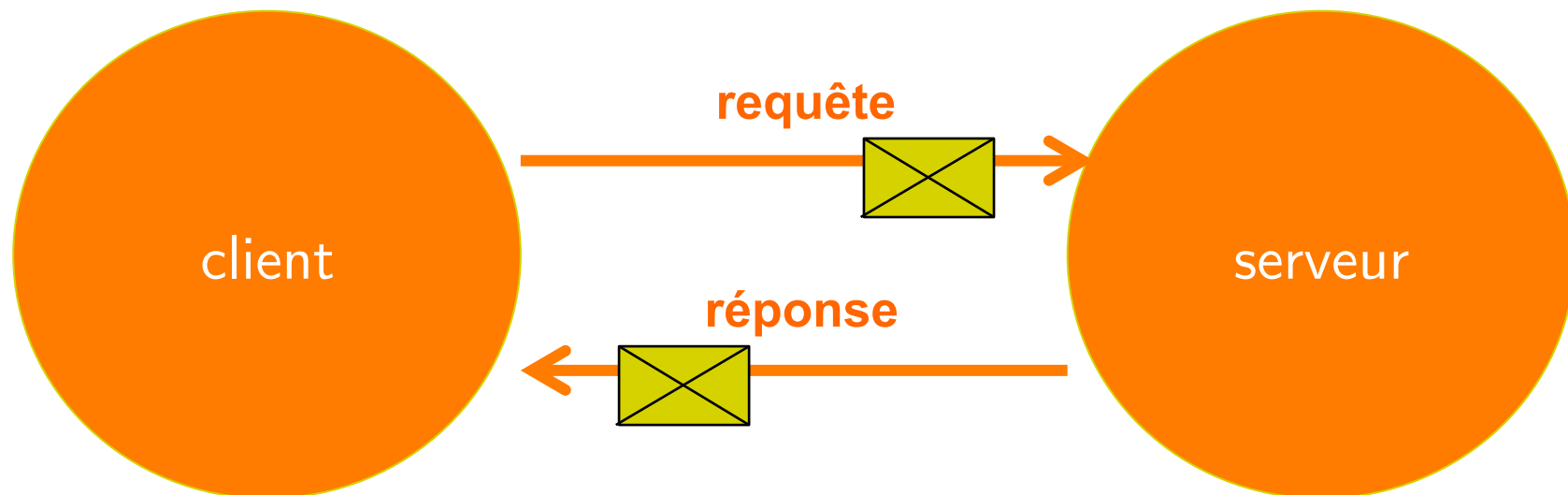


# Architecture client-serveur (répartie) de votre projet

---

## ► Définir le protocole de communication

- quel format de message?
- quelle requête?
- quelle réponse?
- quel enchaînement entre requêtes et réponses?

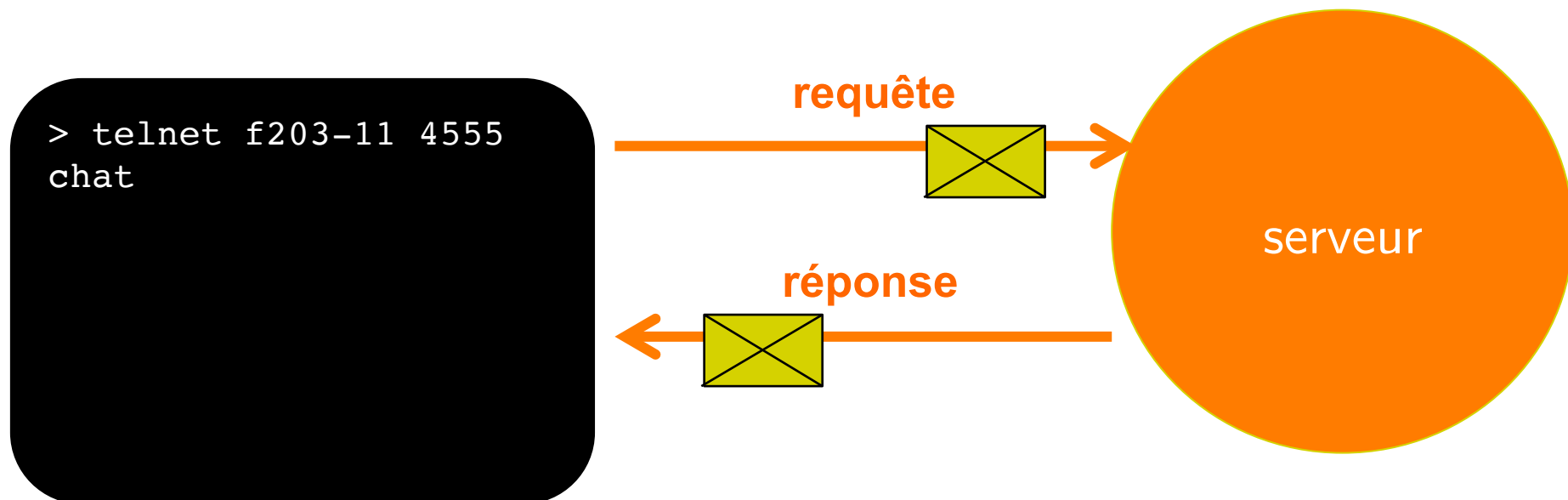


# Architecture client-serveur (répartie) de votre projet

---

## ► Définir le protocole de communication

- Si vous utilisez du "texte", vous pouvez utiliser [telnet](#)
  - lire ligne côté client
  - envoyer au serveur
  - traitement serveur
  - renvoie du texte en résultat



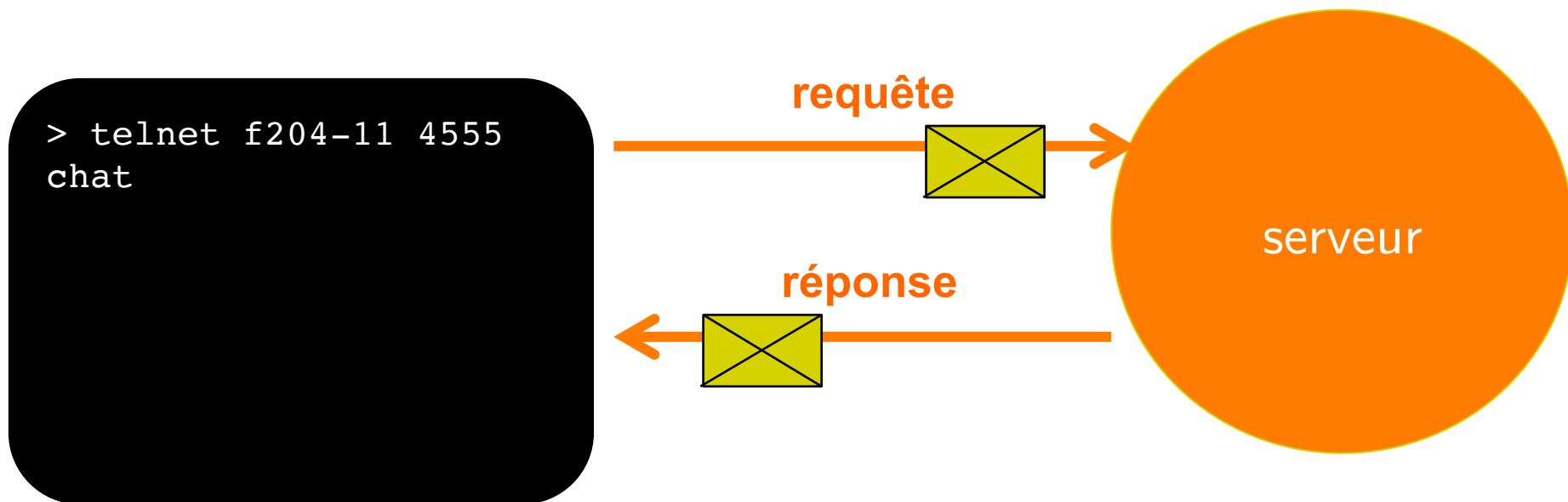


# Architecture client-serveur **ITÉRATIF** de votre projet

---

## ▶ **Un seul processus serveur qui répond**

- ▶ une seule connexion client à la fois
- ▶ une seule requête client à la fois
  - déconnexion après chaque requête ou
  - déconnexion explicite après plusieurs échanges

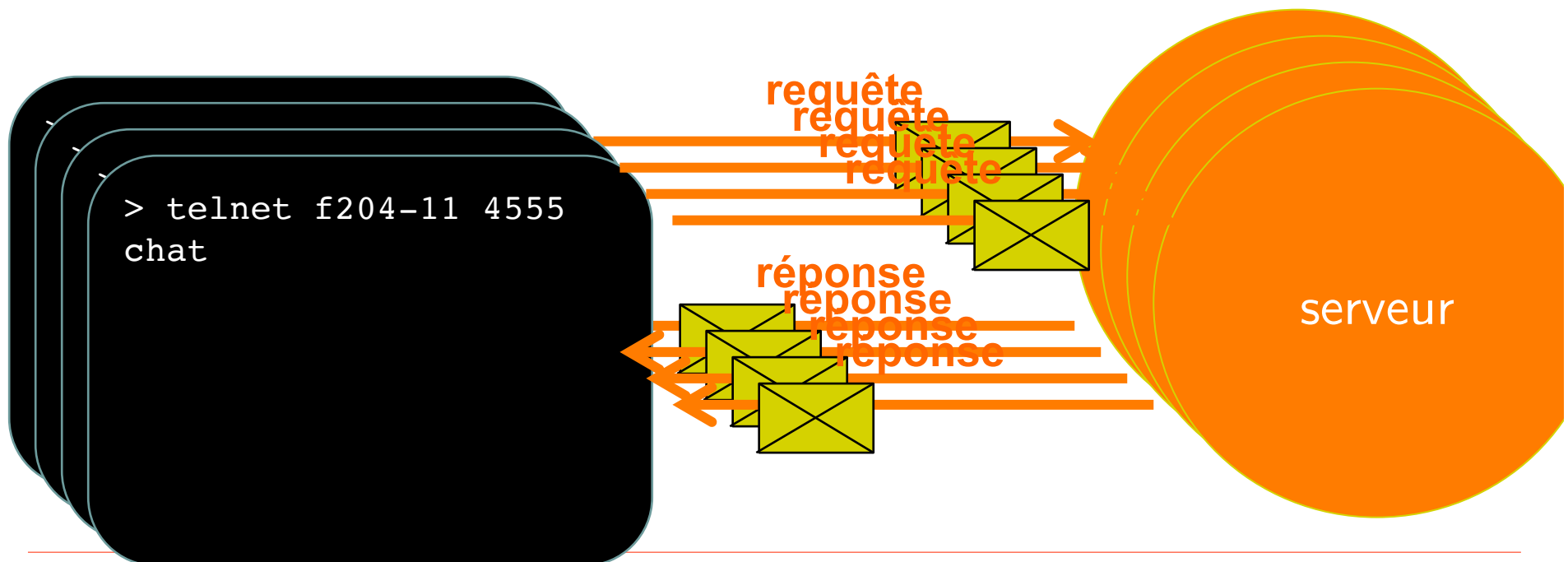


# Architecture client-serveur **CONCURRENT** de votre projet

---

- ▶ **Plusieurs processus serveur (traitants)**

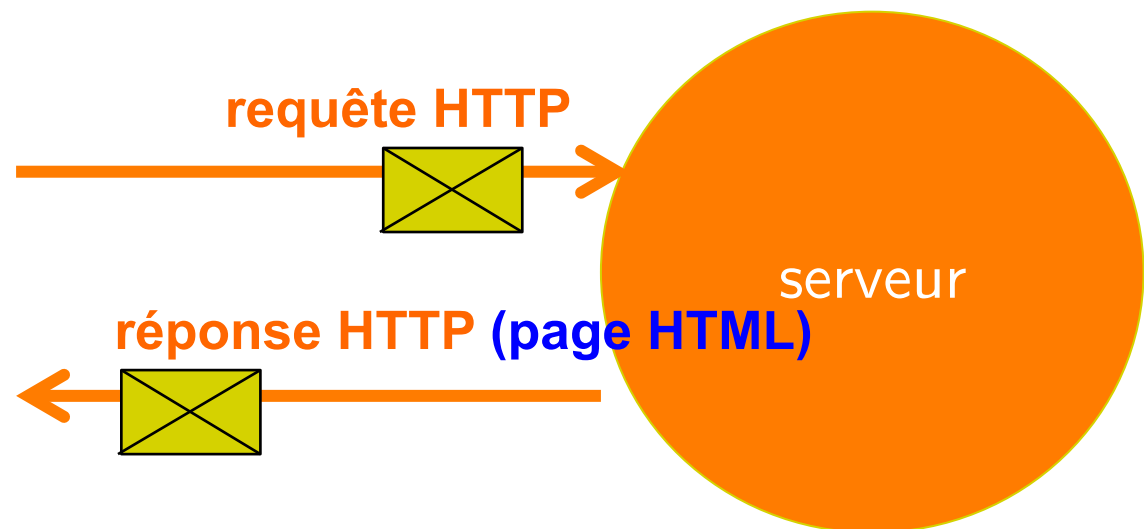
- ▶ plusieurs connexions client à la fois
- ▶ plusieurs requêtes client à la fois



# Architecture client-serveur HTTP

---

- ▶ **Utilisation de HTTP**
  - ▶ règles de protocole imposées



# Architecture client-serveur HTTP : exemple de résultat

---

