

# INFO0503

# Modélisation

---

INTRODUCTION À LA PROGRAMMATION CLIENT / SERVEUR

L3 INFORMATIQUE

OLIVIER FLAUZAC & CYRIL RABAT



# Plan

---

Notion de modèle

Modélisation des acteurs

Modélisation des échanges

Modélisation des exécutions

- Le modèle à message
- Le modèle client / serveur

Etude de cas

# Notion de modèle

---

# Modèle

---

Description détaillée de fonctionnement

Description générale ou particulière

- description d'une application complète
- description des échanges entre deux acteurs · ...

Combinaison de modèles

- Modèle général
- modèle de plus grande taille
- Composants réutilisables

# Pourquoi un modèle ?

---

Définition de tous les cas possibles

- analyse en amont

Définition des fonctionnalités

- définition des erreurs

Normalisation du fonctionnement

Documentation des applications

Réutilisabilité

Conception de solutions techniques associées

- Protocoles

Adaptation aux environnements

# Modèles

---

## Plusieurs modèles

- Modèle général d'une application
- Modèle de communication
- Modèle d'échanges
- Modèle d'exécution
- Modèle de données
- ...

**Autant de modèles que l'on souhaite !**

# Processus de modélisation d'une application

---

Éléments à modéliser

Modélisation des acteurs

Modélisation des interactions

Modélisation des échanges entre les acteurs

Modélisation des méthodes et moyens de calcul

Modélisation des données

- internes aux acteurs
- Échangées

Modélisation de la sécurité

...

# Modélisation des acteurs

---



# Acteurs

---

Éléments / entités / processus qui interviennent dans l'application

Éléments / entités / processus qui font exister l'application

Éléments / entités / processus qui échangent des informations

Affectation d'un rôle précis dans l'application

- Stockage
- Traitement
- ...

# Types d'acteurs

---

## Intervenants humains

- Client d'un site marchand
- Opérateur

## Machine

- Ordinateur contenant un navigateur WEB
- Serveur WEB
- Base de données
- Source de données (capteur)
- Serveur de streaming
- Smartphone
- Application spécifique

# Modélisation des acteurs

---

## Processus de modélisation

1. Etape initiale de la conception
2. Recensement de tous les éléments de l'application
3. Attribution des rôles
4. Description de chaque acteur
5. Mise en place d'un schéma d'interactions possibles

# Modélisation des échanges

---

# Echanges entre les acteurs

---

Echange d'informations entre les acteurs · Définition du transit des données

- Permet de préciser la localisation des données
- Permet de fixer la connaissance des formats et des données
- Connaissance des types (classes, traitement ...)
- Première approche de la sécurité
- définition de la criticité des échanges
- Formalisation des interactions entre les acteurs !

# Modélisation des échanges

---

## Flux de données

- extension des relations entre les acteurs
- identification des échanges sans nécessairement formaliser les données
- Séquencement des échanges
- gestion de la précedence des échanges
- gestion de l'ordre des échanges
- Mode d'échanges
- mode de communication
- impact de l'environnement réseau / système

# Echanges directs

---

Communication directe entre les acteurs

Aucun intermédiaire nécessaire

Connaissance mutuelle obligatoire des deux acteurs

Accès obligatoire entre les acteurs



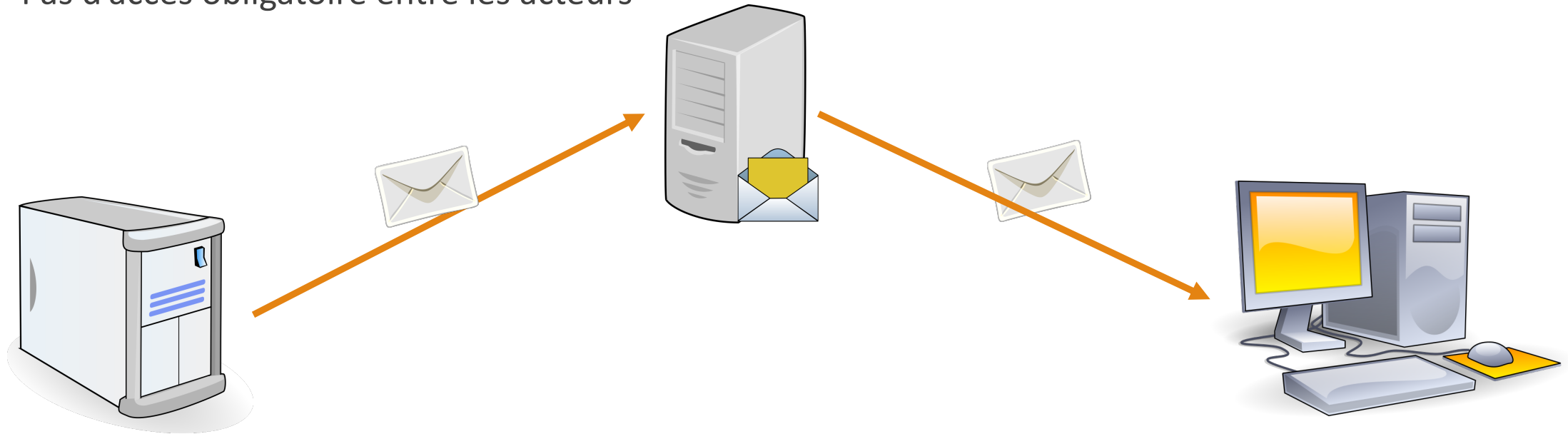
# Echanges indirects

---

Communication via un troisième acteur

Pas de connaissance mutuelle obligatoire des deux acteurs

Pas d'accès obligatoire entre les acteurs





# Flux de données

---

## Que modéliser

- Transport d'informations
- Transit des données
- Evaluation de la volumétrie des données échangées
- taille des données
- fréquence d'échanges

# Séquences d'échanges

---

## Types d'échanges

- Echanges simples
- Echange ne nécessitant pas de réponse
- Juste un transfert d'informations
- Echanges enchainés
- Suite d'échanges
- Question / Réponse
- Série entre différents acteurs

# Echanges simples

---

## Transit d'un message

- Pas de réponse attendue
- Envoie d'un mail
- Envoie d'un SMS
- Peut toutefois amener une réponse sur le même modèle

# Echanges enchaînés

---

## Schémas possibles

- Question / Réponse
- Requête à un annuaire, une base de données
- Série de requêtes
- Enchaînement commande / paiement / accord d'un organisme financier

# Modes d'échanges

---

Que modéliser ?

- Initiation des communications
- Gestion de la signalisation
- Gestion du transport
- Éléments de gestion système des échanges

# Modes d'échanges connectés

---

## Principe

- Etablissement d'un canal de communication entre les acteurs
- Vérification de la présence de la destination
- Communication bidirectionnelle dans le canal
- Maintenance de la connexion entre les acteurs

Non lié à un modèle d'exécution

# TCP

---

## Principe

- Protocole fiable
- Gestion système de la distribution des messages
- Gestion de la synchronisation
- Gestion de l'intégrité des données
- Socket TCP
- Indépendant des systèmes et langages

# Modes d'échange non connectés

---

## Principe

- Communications légères
- Pas de canal fixe de communication · Envoi aveugle
- Léger et efficace
- Gestion déléguée au développeur

Non lié à un modèle d'exécution



# UDP

---

## Principe

- Protocole non fiable
- Absence de contrôle
  - De présence
  - De distribution
  - D'intégrité du contenu
- Datagrammes UDP
- Indépendant des systèmes et langages

# Modélisation des exécutions

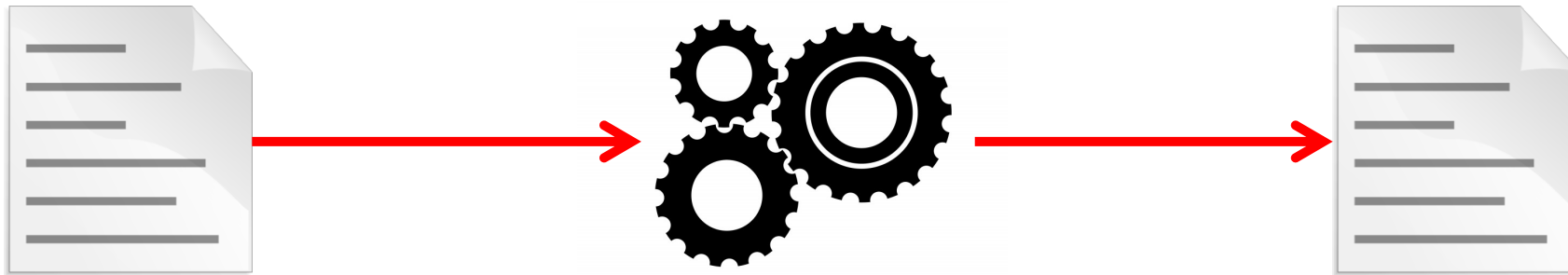
---

# Une exécution ?

---

Données en entrée (brute ou extraction)

- Traitement
- Données en sortie (brute ou insertion)
- Nécessité de localiser les différents éléments



# Exécution locale

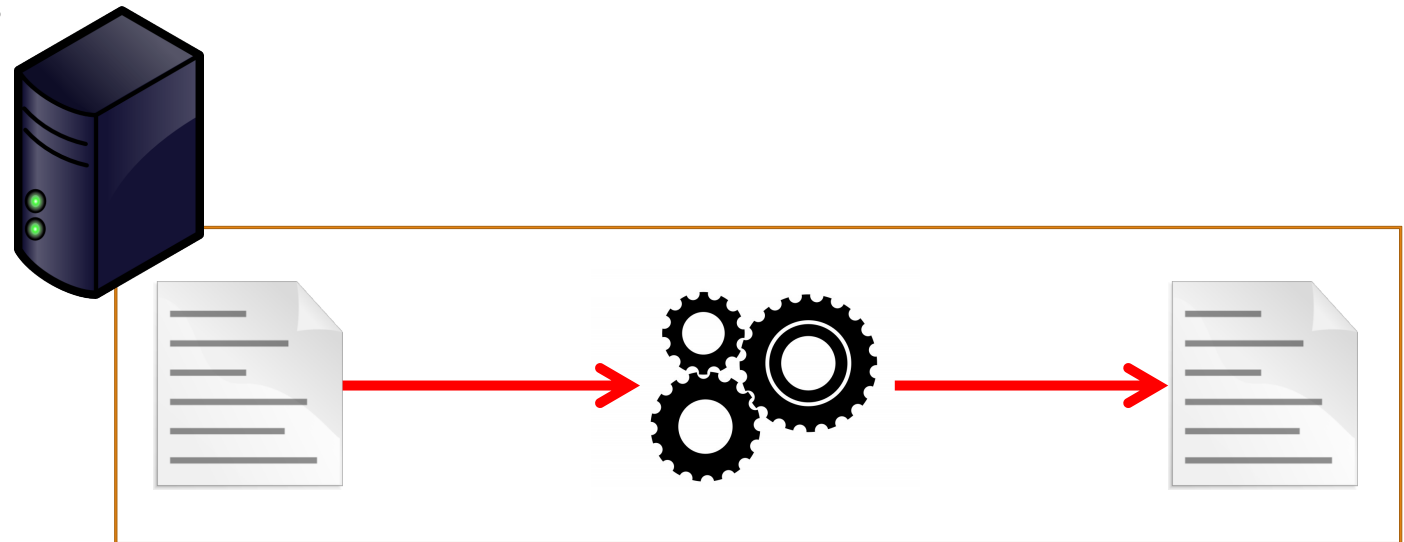
---

Traitements et données sur le même acteur

Schéma simple

Limité :

- aux connaissances locales
- aux capacités locales : mémoire, calcul ...



# Exécution répartie

---

Traitements et données repartis sur différents acteurs

Toutes les combinaisons sont possibles

Urbanisation de l'application

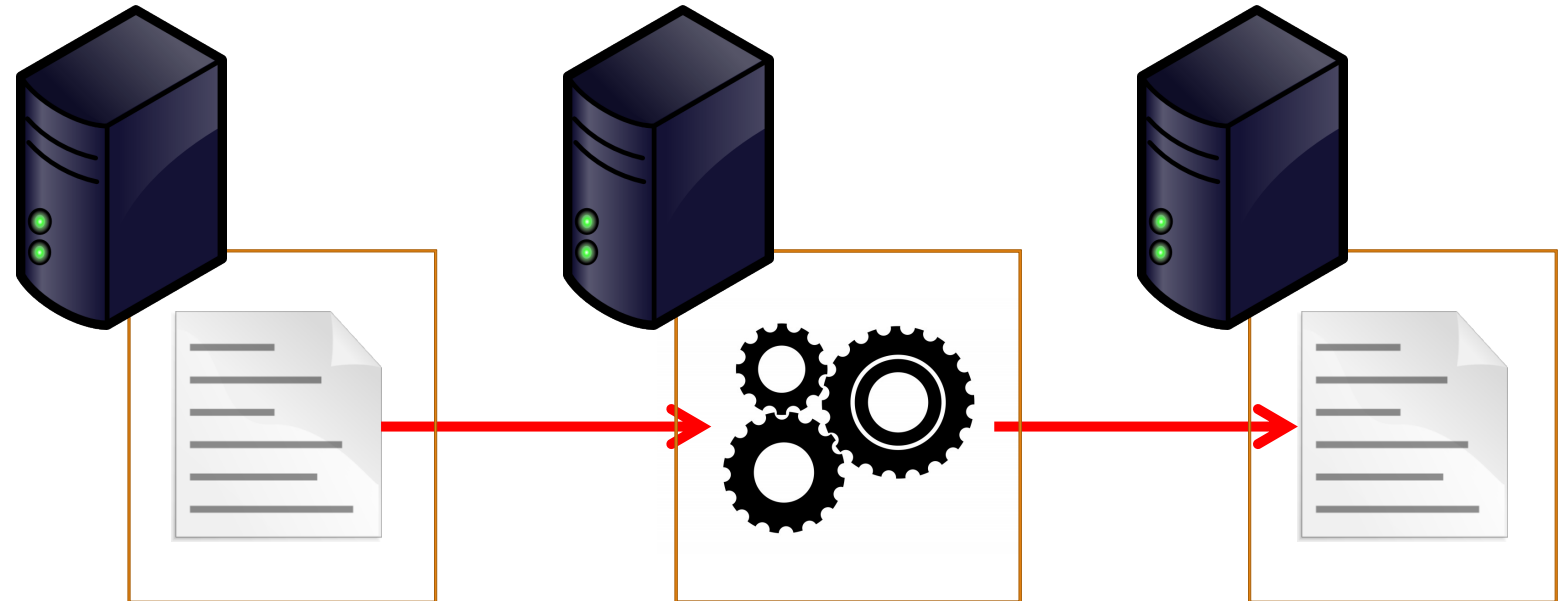
- Définition de l'implantation des acteurs
- Définition de l'implantation du code

# Urbanisation : trois acteurs

---

Un acteur pour chaque élément

- Communication entre chaque étape



# Urbanisation : deux acteurs

---

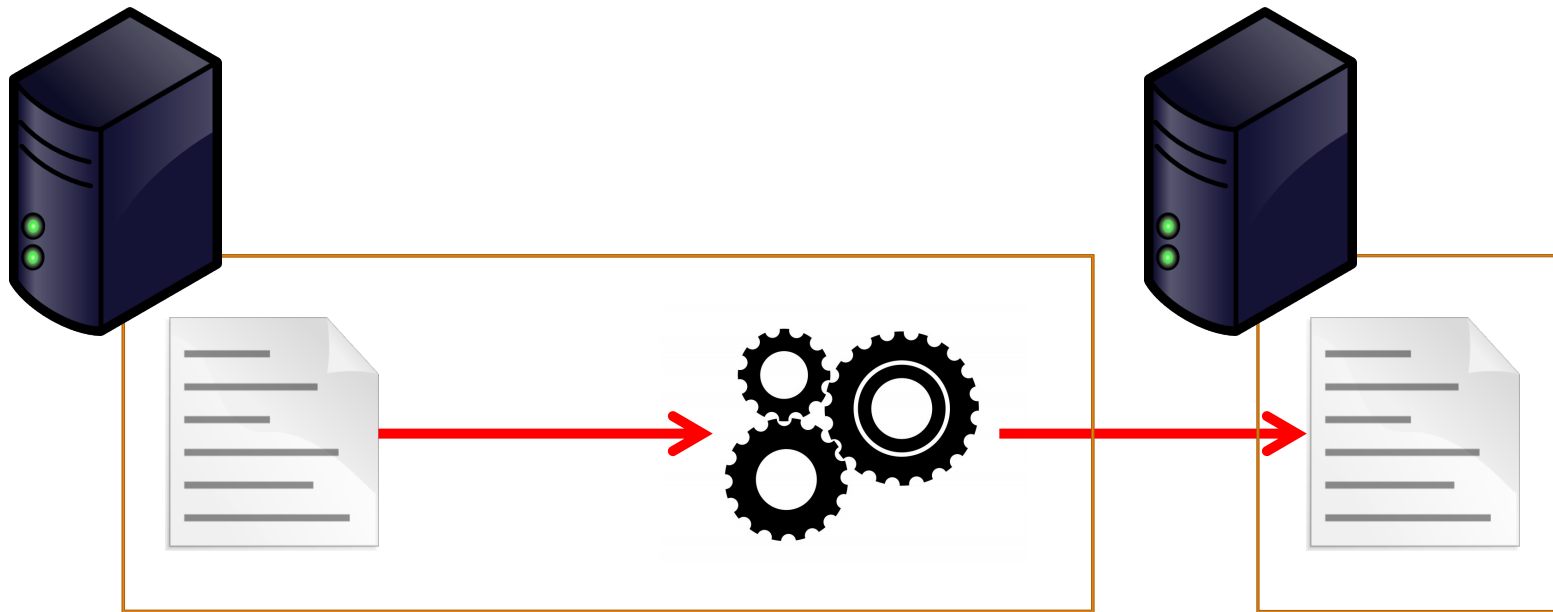
3 combinaisons possibles

Communication entre les acteurs

Liaison des acteurs 2 à 2

# Urbanisation : deux acteurs

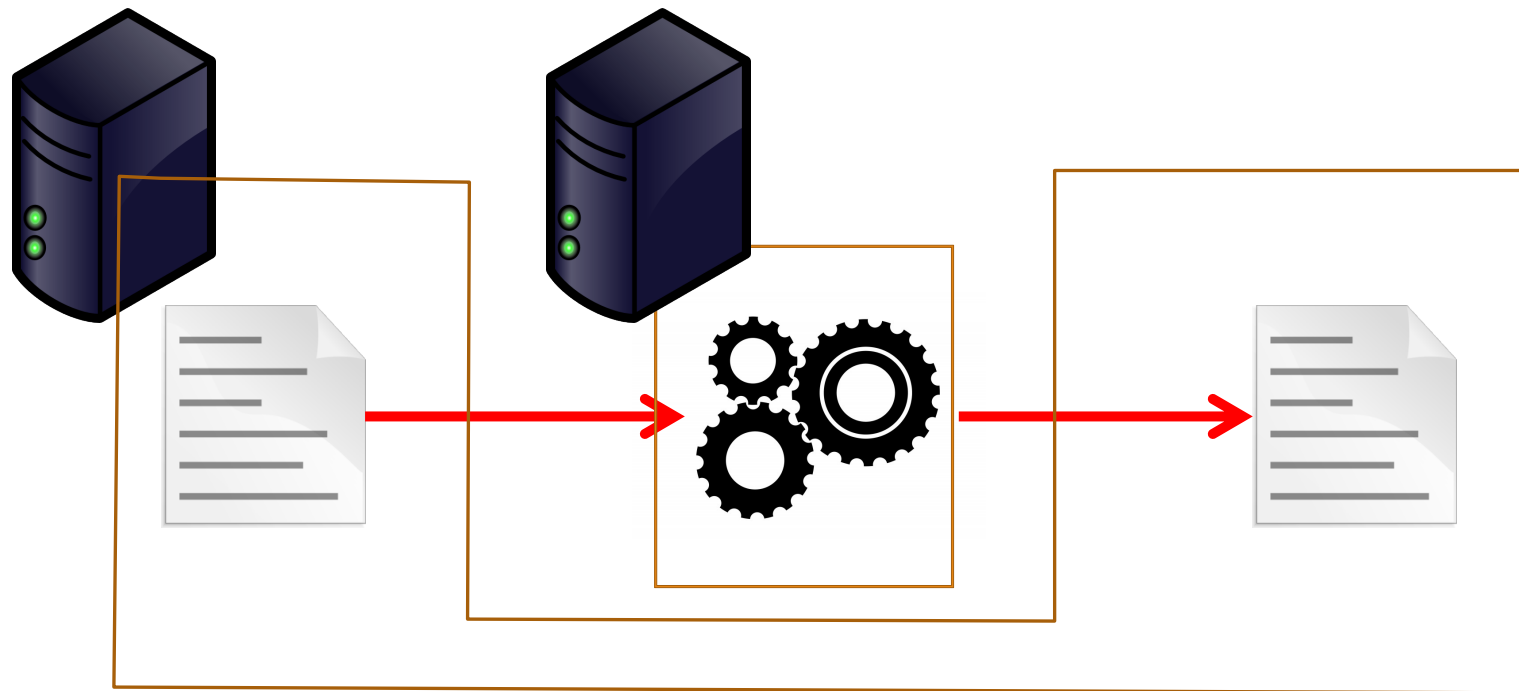
---





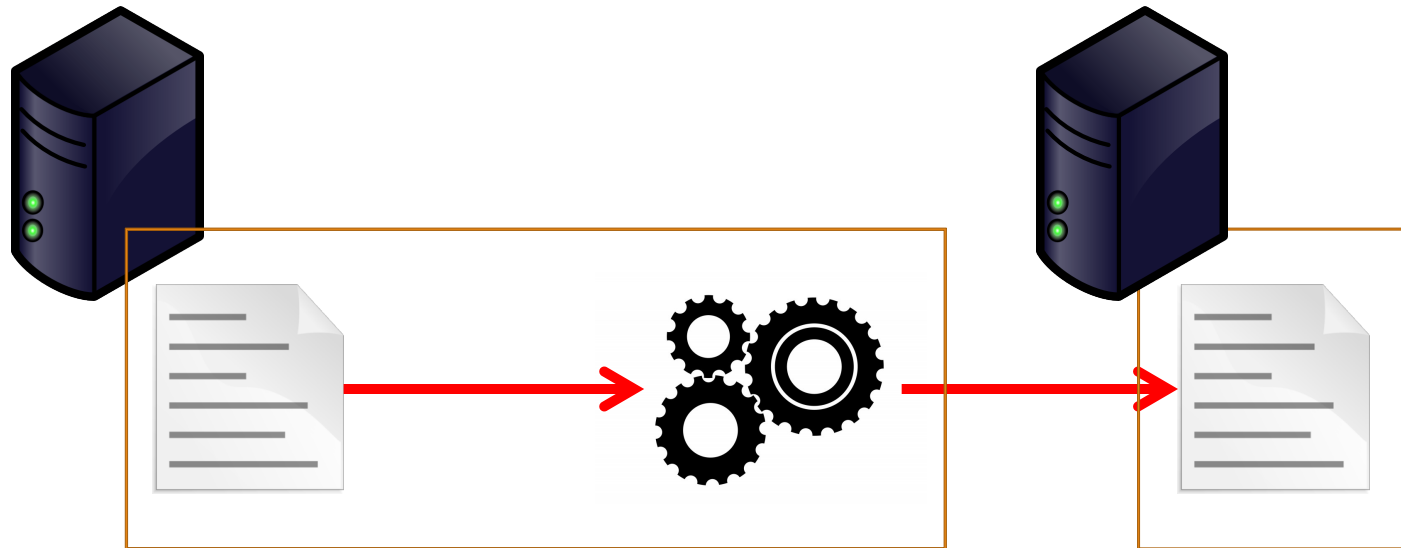
# Urbanisation : deux acteurs

---



# Urbanisation : deux acteurs

---



# Pourquoi des exécutions reparties

---

## Avantages

- Séparation des données et des traitements
- Exploitation des capacités des machines
- Spécialisation des machines

# Modèle à message : définition

---

## Principe

- Modèle basé sur l'échange d'un message
- Organisation
- Données sources émises par un acteur et traitement et données résultat sur un autre acteur
- Traitement et données sur un acteur, données résultats émises à un autre acteur
- Pas de réponse requise

# Modèle à message : exemples

---

## Services classiques

- Enregistrement des logs sur une machine distante
- Serveur d'impression

## Applications liées aux réseaux sociaux

- Dépôt d'une information
- Envoie d'un texte / image ...
- Données sources sur l'acteur utilisateur
- Traitement et données résultats sur l'acteur réseau social
- Notification
- Réception des infos d'un abonné / ami
- Données sources et traitement sur l'acteur réseau social
- Données résultats sur l'acteur utilisateur

# modèle client / serveur

---

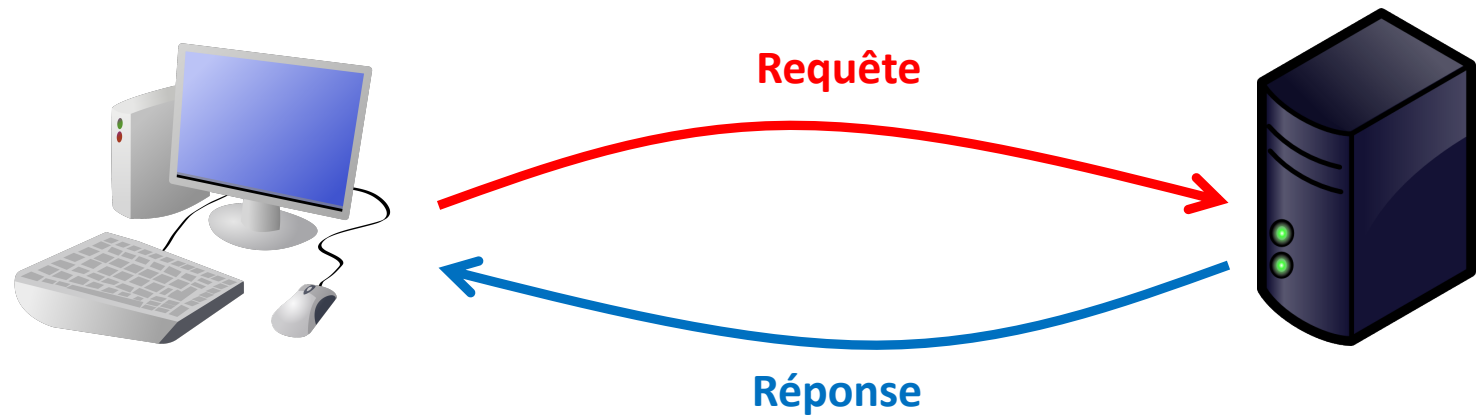
Définition des échanges et des exécutions

Mise en œuvre avec 2 acteurs

- Le client
- Le serveur

Modèle basé sur l'échange de deux messages

- Une requête
- Une réponse



# Modèle client / serveur : le client

---

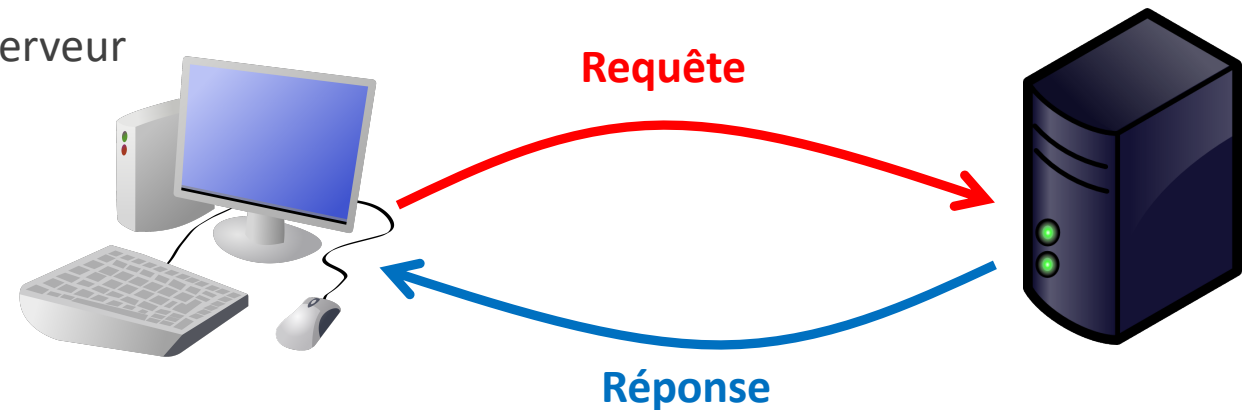
## Rôle du client

- Demandeur de service
- Emission d'une requête

## Connaissance de la ressource

- Adresse
- Type d'accès
- Connaissance des moyens d'échange avec le serveur
  - Normalisation des données
  - Protocole d'échange

## Consommation de la réponse associée

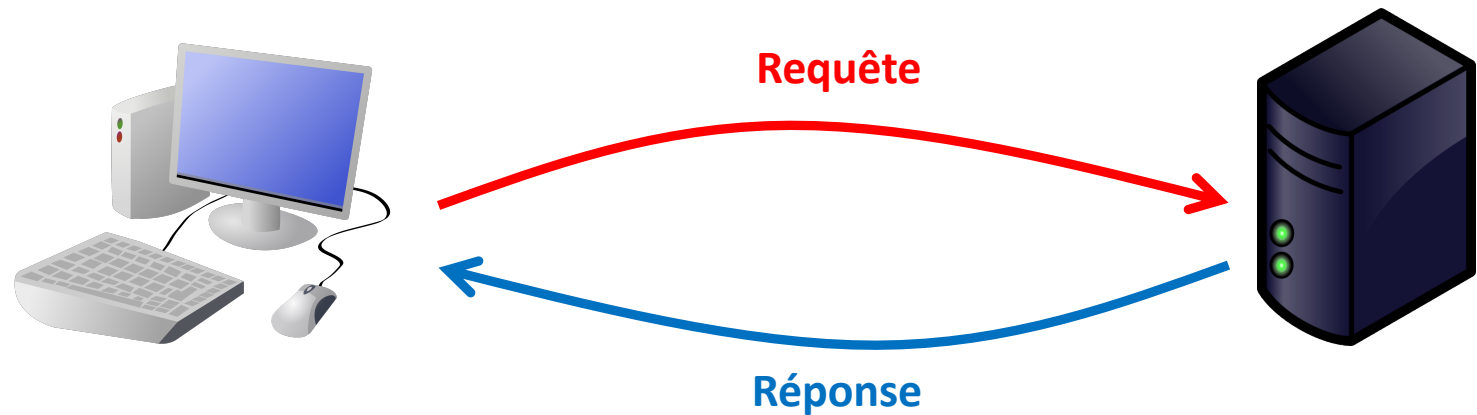


# Modèle client / serveur : le serveur

---

## Rôle du serveur

- Mise à disposition d'un service
- Réception de la requête
- Traitement de la requête
- Envoi de la réponse





# Modèle client / serveur : exemple

---

Services internet

Serveur WEB

Serveur DNS

Serveur

Envoi de la réponse

# Etude de cas

---

# Une application de *tracking* sportif

---

Le *tracking* d'activités consiste à collecter périodiquement différentes informations au cours d'une activité (jogging, vélo, ski, chasse au dahu ...). Ces informations sont alors stockées tout au long de l'activité.

Dans notre application, le smartphone d'un sportif émettra périodiquement des positions GPS à destination d'une machine stockant ces différentes positions. Une partie de l'application dédiée au traçage de courbes et aux statistiques exploitera les données ainsi enregistrées.

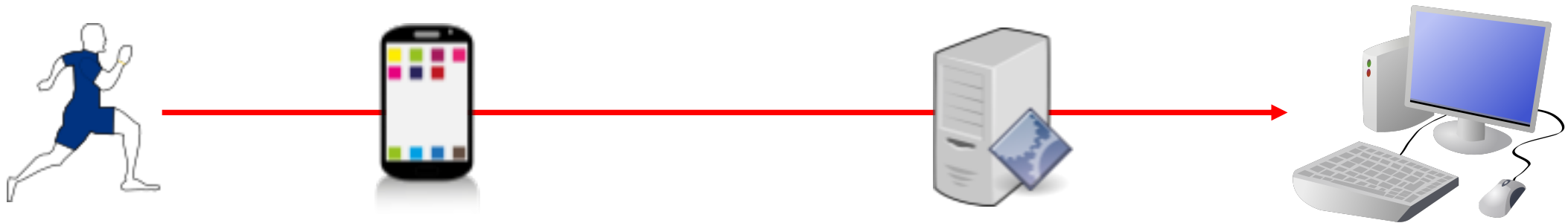
Une activité sportive est identifiée par : l'identifiant du sportif, l'heure de début et l'heure de fin, le sport pratiqué, une liste de points GPS correspondant aux points de passage, ainsi que les heures associées.

# Identification des acteurs

---

## Les acteurs

- Un acteur humain : Le sportif
- Un acteur numérique : terminal de type smartphone
- Un acteur numérique : machine de sauvegarde des activités
- Un acteur numérique : machine traçant les courbes et réalisant les calculs



# Modélisation des échanges

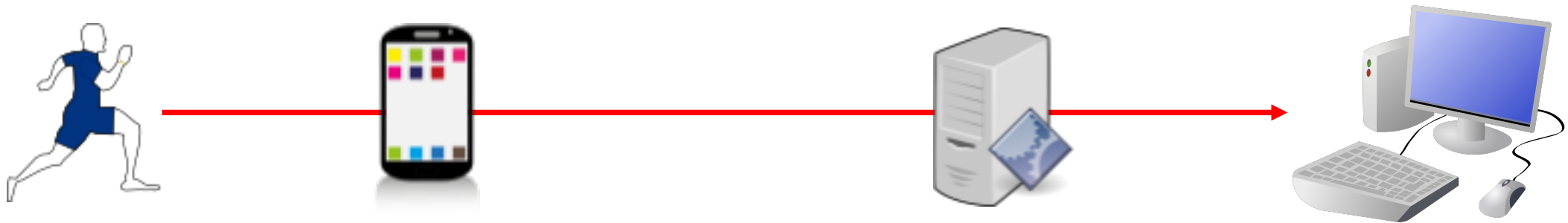
---

Phase 1 : Smartphone → sauvegarde · échange des données GPS

- Emission périodique

Phase 2 : Sauvegarde → visualisation

- Récupération des données d'une activité complète
- Récupération à la demande du visualiser



# Modélisation des exécutions : phase 1

---

## Sauvegarde des informations

- Modèle à passage de messages
- Pas de requête / réponse

## Opération sur la sauvegarde

### 1. Début de la séance

- Création de l'activité
- Enregistrement de la position initiale

### 2. Réception d'une position

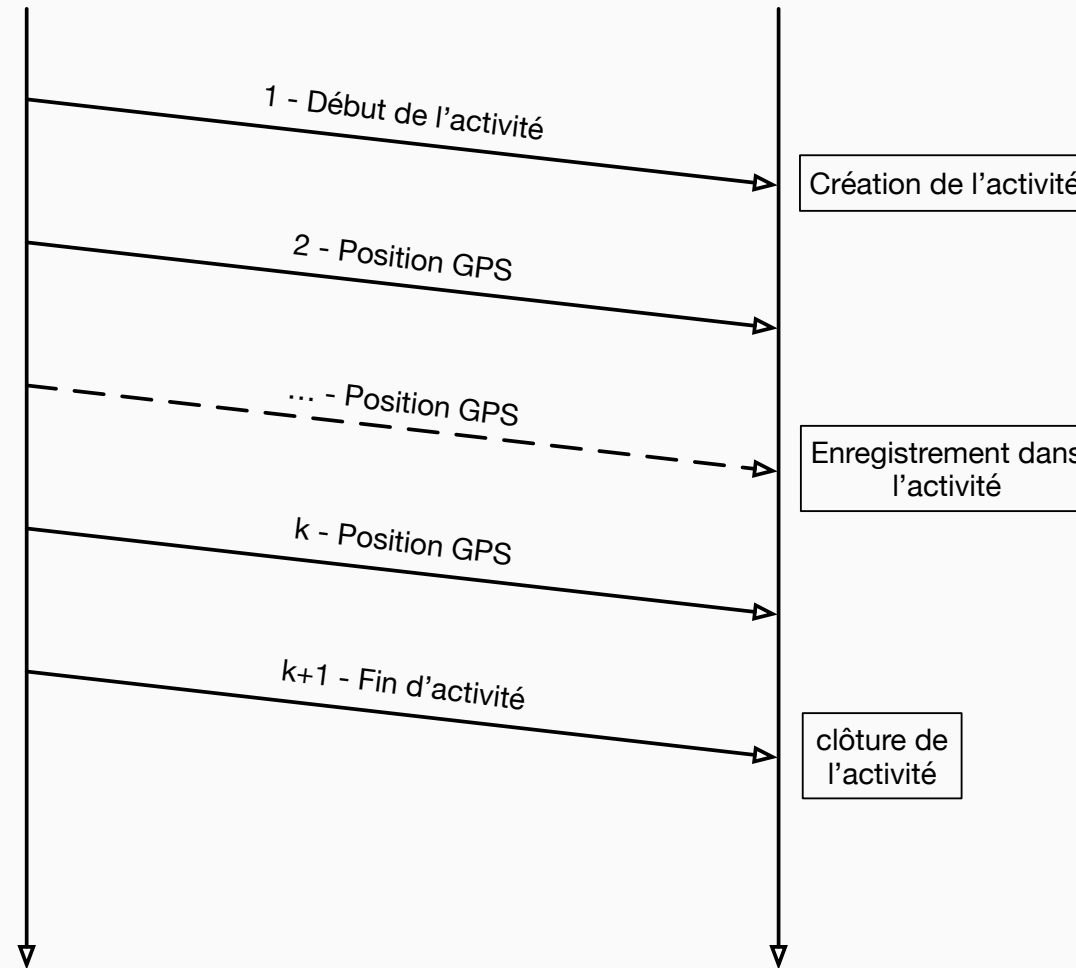
- Ajout de la position à l'activité en cours

### 3. fin de la séance

- Enregistrement de la dernière position
- Clôture de l'activité

Smartphone

Sauvegarde



# Modélisation des exécutions : phase 2

---

## Visualisation des informations

- Modèle client / serveur

## Requête de demande d'une activité précise

### 1. Requête de demande d'une activité

- Recherche de l'activité
- Emission de l'activité
- Erreur si l'activité n'existe pas

### 2. Réception de l'activité

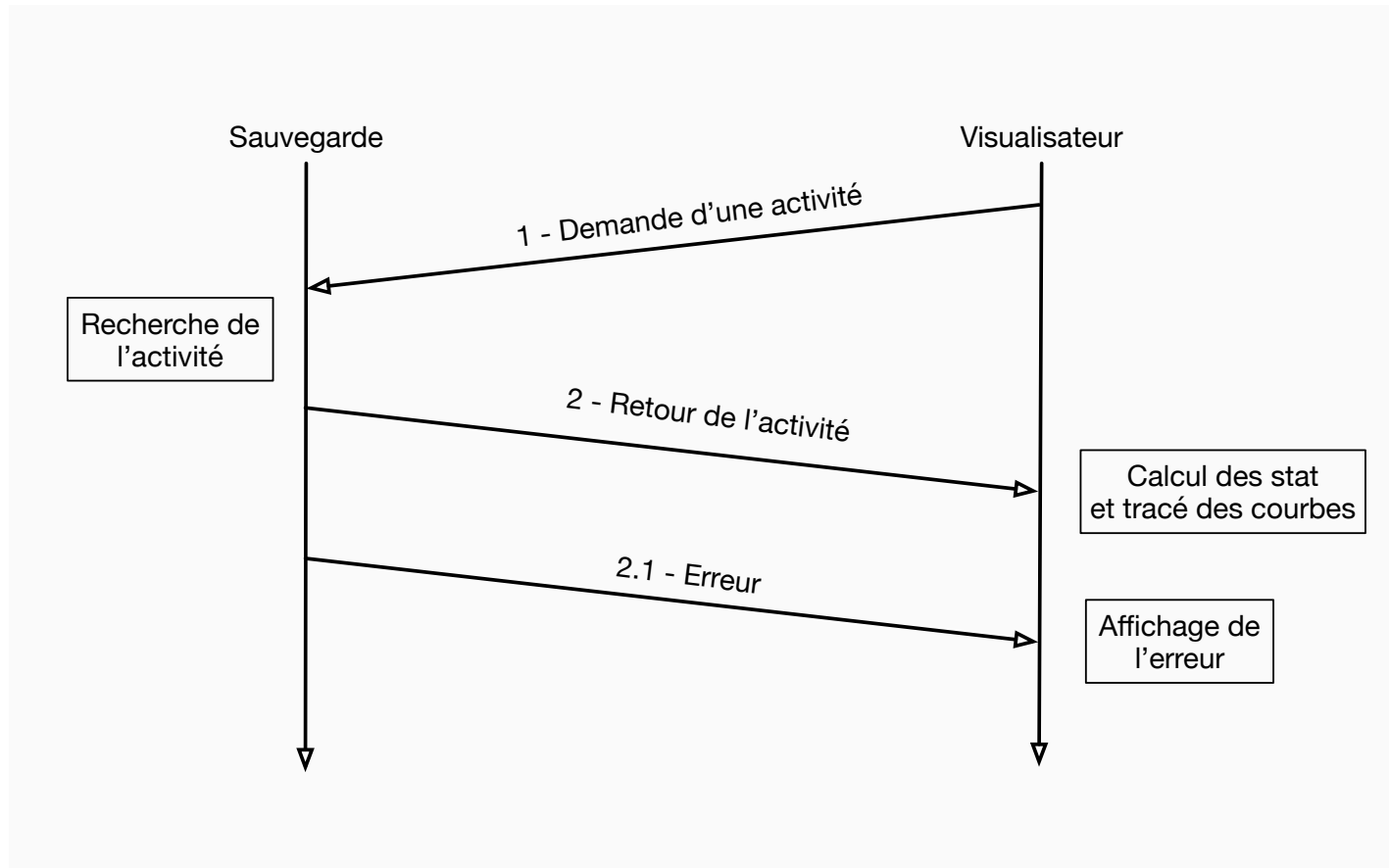
- Calcul et traçage des courbes

### 3. Réception d'une erreur

- informer l'utilisateur



# Exécutions : phase 2



# Reste à modéliser

---

## Données

- Représentation des données
- Format des données

## Echanges et protocole

- Connecté ou non connecté
- Quelle solution de transport ?

# Pour aller plus loin

---

## Choix technologiques

- Plates-formes
- Langages

## Et les erreurs ?

- Pas de message de début ou de fin d'activité
- Panne d'un des acteurs
- ...

# Pas de solution unique !

---

