

## TME 2

Dans cette partie du TP, vous allez échanger des demandes / messages entre deux serveurs différents. Pour chaque demande nous allons créer une Lettre et l'envoyer au serveur distant pour l'exécuter.

### Processus :

Une personne veut envoyer plusieurs demandes (Lettres). Les demandes doivent être traitées le plus vite possible, pour cela on doit attribuer le travail sur plusieurs worker-server.

Vous allez prendre en entrée un fichier, contenant plusieurs lignes où chaque ligne correspond à une demande.

1. Lire le fichier
2. Pour chaque ligne créer une Lettre (faire une sauvegarde dans la base de données)
3. Envoyer la demande (envoyer un message contenant l'objet Lettre)
4. Afficher la demande / (l'écrire dans un fichier / la sauvegarder dans la base de données)  
: La demande "....." est traitée

Pour cela vous devez créer une tâche et le(s) étapes correspondantes :

1. Créer un nouveau Job "letterJob" qui contient le step "letterStep"
2. Créer un nouveau Step "letterStep" avec un Tasklet ou / et un itemRead / itemProcessor / itemWriter

Pour simuler un traitement de la demande vous pouvez utiliser un thread sleep.

### Sink / Processor

Création d'un Processus de Récupération de Messages

1. Ajouter une méthode dans le worker pour écouter un flux de messages :  
Dans le worker, ajoutez une méthode annotée avec `@StreamListener` pour écouter les messages du flux. Utilisez les annotations Spring Cloud Stream.
2. Lire les messages avec le header `custom_info='start'` et les transformer :  
La méthode doit lire les messages qui ont le header `custom_info='start'`, les transformer, et les renvoyer avec le header `custom_info='end'`.
3. Ajouter des logs pour suivre le flux des messages :

Pour finaliser le projet, vous devez implémenter soit le "Remote Chunking" ou le "Remote Partitioning".