

# Message broker et programmation asynchrone - Symfony Messenger

## Objectif

L'objectif de ce TP est de créer un système de gestion des tâches asynchrones en utilisant Symfony Messenger. Vous allez créer un système pour enregistrer des tâches à faire et les traiter de manière asynchrone. Vous allez également apprendre à utiliser des handlers pour traiter les messages et à configurer des middleware pour intercepter et modifier le comportement de Symfony Messenger.

## Étapes

### Étape 1 : Installation

Commencez par installer Symfony Messenger en utilisant Composer. Vous pouvez suivre les instructions fournies sur la page de documentation de Symfony Messenger.

### Étape 2 : Configuration

Configurez Symfony Messenger en utilisant un fichier de configuration YAML. Définissez un bus par défaut pour l'envoi et le traitement des messages.

### Étape 3 : Création d'un message

Créez une classe de message simple qui contient une propriété `task` pour représenter une tâche à faire. Cette classe doit implémenter l'interface `MessageInterface`.

### Étape 4 : Traitement des messages

Créez un handler pour traiter les messages de tâches. Le handler doit recevoir un message de tâche et effectuer une tâche simulée, comme l'écriture du contenu de la tâche dans un fichier. Utilisez l'annotation `@Handler` pour définir le handler.

### Étape 5 : Envoi de messages

Créez une action de contrôleur pour enregistrer une tâche à faire. L'action doit créer un message de tâche et l'envoyer à travers le bus par défaut. Utilisez l'objet `MessageBusInterface` pour envoyer le message.

### Étape 6 : Middleware

Configurez un middleware pour intercepter les messages avant qu'ils ne soient traités par les handlers. Utilisez le middleware pour ajouter une propriété `created_at` au message et définir la date et l'heure actuelles.

## Étape 7 : Routage de messages

Créez un second handler pour traiter les messages de tâches qui ont une priorité élevée. Ce handler doit effectuer une tâche différente de celle du premier handler. Utilisez l'annotation `@Handler` pour définir le handler.

## Étape 8 : Débogage

Utilisez l'outil de débogage de Symfony pour surveiller les messages envoyés et traités. Vérifiez que les messages sont correctement envoyés et traités par les handlers.

## Conclusion

Ce TP vous a montré comment utiliser Symfony Messenger pour gérer des tâches asynchrones dans une application Symfony. Vous avez appris à créer des messages, à les traiter en utilisant des handlers et à configurer des middleware pour intercepter et modifier le comportement de Symfony Messenger. Vous avez également appris à utiliser l'outil de débogage de Symfony pour surveiller les messages envoyés et traités.