



TP 3 : Développement d'Agents avec SPADE

Objectif : Ce TP vise à approfondir la compréhension et les compétences des étudiants dans le développement d'applications multi-agents en Python, en utilisant SPADE pour créer un agent simple et établir une communication de base entre agents.

Partie 1 : Création d'un Agent Simple

Dans cette première partie, les étudiants apprendront à créer un agent de base qui affiche un message dans la console. Cet exercice introduit la structure de base d'un agent SPADE et montre comment implémenter un comportement cyclique.

Instructions

1. Installer SPADE si ce n'est pas déjà fait : `pip install spade`.
2. Utiliser le code fourni pour créer un fichier Python, par exemple `my_first_agent.py`.
3. Exécuter le script pour observer le comportement de l'agent. L'agent doit afficher un message de démarrage, suivi d'un message cyclique indiquant qu'il est en cours d'exécution.

Partie 2 : Communication entre Agents

Dans la deuxième partie, les étudiants mettront en place une communication simple entre deux agents : un émetteur (`PeriodicSenderAgent`) et un récepteur (`ReceiverAgent`). L'émetteur enverra des messages périodiquement au récepteur, qui affichera le contenu des messages reçus.

Instructions

1. Créer deux fichiers Python : `sender_agent.py` et `receiver_agent.py`, ou intégrer les deux classes dans un seul script si préféré.
2. Adapter le code fourni pour que l'`InformBehav` du `PeriodicSenderAgent` envoie des messages à `ReceiverAgent`.
3. Assurez-vous que les JID (Jabber Identifiers) et les mots de passe sont correctement configurés pour vos agents.
4. Exécuter les agents et observer la communication entre eux. Le récepteur doit afficher le contenu des messages envoyés par l'émetteur.

Conseils et Bonnes Pratiques

- **SPADE Documentation:** [SPADE Read the Docs](#).
- **Gestion des Identifiants XMPP :** Assurez-vous d'utiliser des identifiants XMPP valides pour vos agents. Vous pouvez utiliser des serveurs XMPP publics pour créer des comptes si nécessaire (par exemple : [anonym.im](#)).
- **Debugging :** Utilisez des outils de logging ou des points d'arrêt pour debugger le comportement de vos agents si vous rencontrez des difficultés.
- **Sécurité :** Soyez conscient des implications de sécurité lors de l'utilisation de serveurs XMPP, en particulier avec des mots de passe et des communications sensibles.