Thomas Recouvreux, Agathe Oddon

IA04 - Compte-rendu du TP n°3

Questions :

1. ChatAgent intéragit avec les autres agents en receptionnant leurs messages et en envoyant.

ChatAgentFrame affiche les messages qu'il reçoit dans une fenêtre JFrame.

MultAgent effectue le produit de deux chiffres.

**2. Quelles sont les compétences (skill) de chaque agent ?**

ChatAgent a deux behaviour : ChatReceiverBehaviour et ChatMultRecvBehaviour.

ChatReceiverBehaviour :

N'écoute que les messages avec ReplyWith == 'chat'. Lorsqu'un message est reçu, il est envoyé vers la GUI, puis si le message ressemble à une multiplication, par exemple « 3 x 4 ? », alors un nouveau message est envoyé à MultAgent.

ChatMultRecvBehaviour :

N'écoute que les messages avec ReplyWith == 'mult'. Lorsqu'un message est reçu, il s'agit d'une réponse à une multiplication, le message est simplement retransmit à tous les ChatAgents.

MultAgent a le même behaviour que dans le TD précédent, c'est à dire :

* Quand un message est reçu :
  + Si le message n'est pas null et qu'il contient "x"
    - Le produit des deux chiffres est calculé.
    - Cette valeur est placée dans la réponse au message, qui est envoyée.
  + Sinon le behaviour est bloqué

4. ChatAgent envoie un message de type INFORM pou chatter avec les autres agents, et un REQUEST pour demander le calcul d'un produit à MultAgent.

MultAgent répond dans un message de type INFORM si il réussit à effectuer le calcul. Sinon, il envoie une FAILURE.

**5. Quel est le langage de contenu des messages échangés: syntaxe (basée sur le format JSON) et interprétation par les agents. Le dictionnaire du système : ensemble du vocabulaire utilisé dans le système et des définitions**

Les messages concernants le chat sont envoyé en text brut, les ChatAgent récupèrent l'expéditeur grâce à la méthode message.getSender().

Les messages pour la multiplication sont envoyées sous forme textuel : « nombre x nombre » la réponse est un nombre sous forme textuelle aussi.

**6. Comment sont échangés les messages ? On pourra faire des diagrammes style**

**diagramme de séquences UML mais basés sur des agents et l'interface graphique. --> Je sais pas si mon diagramme correspond vraiment, j'ai mis les utilisateurs..**

Code pour le diagramme:

ChatAgent1->ChatAgentFrame1:création

ChatAgent2->ChatAgentFrame2:création

Utilisateur1->ChatAgentFrame1: actionPerformed : tappe "coucou"

ChatAgentFrame1->ChatAgent1: GUI Event : param="coucou"

alt

ChatAgent1->ChatAgent1: INFORM:"coucou"

ChatAgent1->ChatAgentFrame1: Event new line("coucou")

note right of ChatAgentFrame1: affiche "coucou"

else

ChatAgent1->ChatAgent2:INFORM:"coucou"

ChatAgent2->ChatAgentFrame2: Event new line("coucou")

note right of ChatAgentFrame2: affiche "coucou"

end

Utilisateur2->ChatAgentFrame2: actionPerformed : tappe "6x4?"

ChatAgentFrame2->ChatAgent2: GUI Event : param "6x4?"

ChatAgent2->ChatAgent1:INFORM:"6x4?"

ChatAgent1->ChatAgentFrame1:Event new line("6x4?")

ChatAgent1->MultAgent1:REQUEST:"6x4"

MultAgent1->ChatAgent1:INFORM:"24"

alt

ChatAgent1->ChatAgent1:INFORM:"24"

ChatAgent1->ChatAgentFrame1:Event new line("24")

else

ChatAgent1->ChatAgent2:INFORM:"24"

ChatAgent2->ChatAgentFrame2:Event new line("24")

end

ChatAgent utilise la fonction getMultiplicator dans ChatReceiverBehaviour pour connaître les agents multiplicateurs disponibles. Pour trouver les autres agents, il utilise getAgents().

MultAgent ne fait que répondre à l'emetteur du message qu'il a reçu, ilt n'a donc pas besoin de connaître la liste des agents.