

Systemy Agentowe

Laboratorium 5

Celem zadania jest zapoznanie się procesem prowadzenia negocjacji w środowisku wieloagentowym. Przy realizacji zadania należy posłużyć się przykładem (Alchemists) zaprezentowanym na wykładzie lub projektem z poprzedniego zadania.

Zawartość przykładu

Szczegółowy opis projektu, agentów, zachowań i ontologii znajduje się w instrukcji do laboratorium 4.

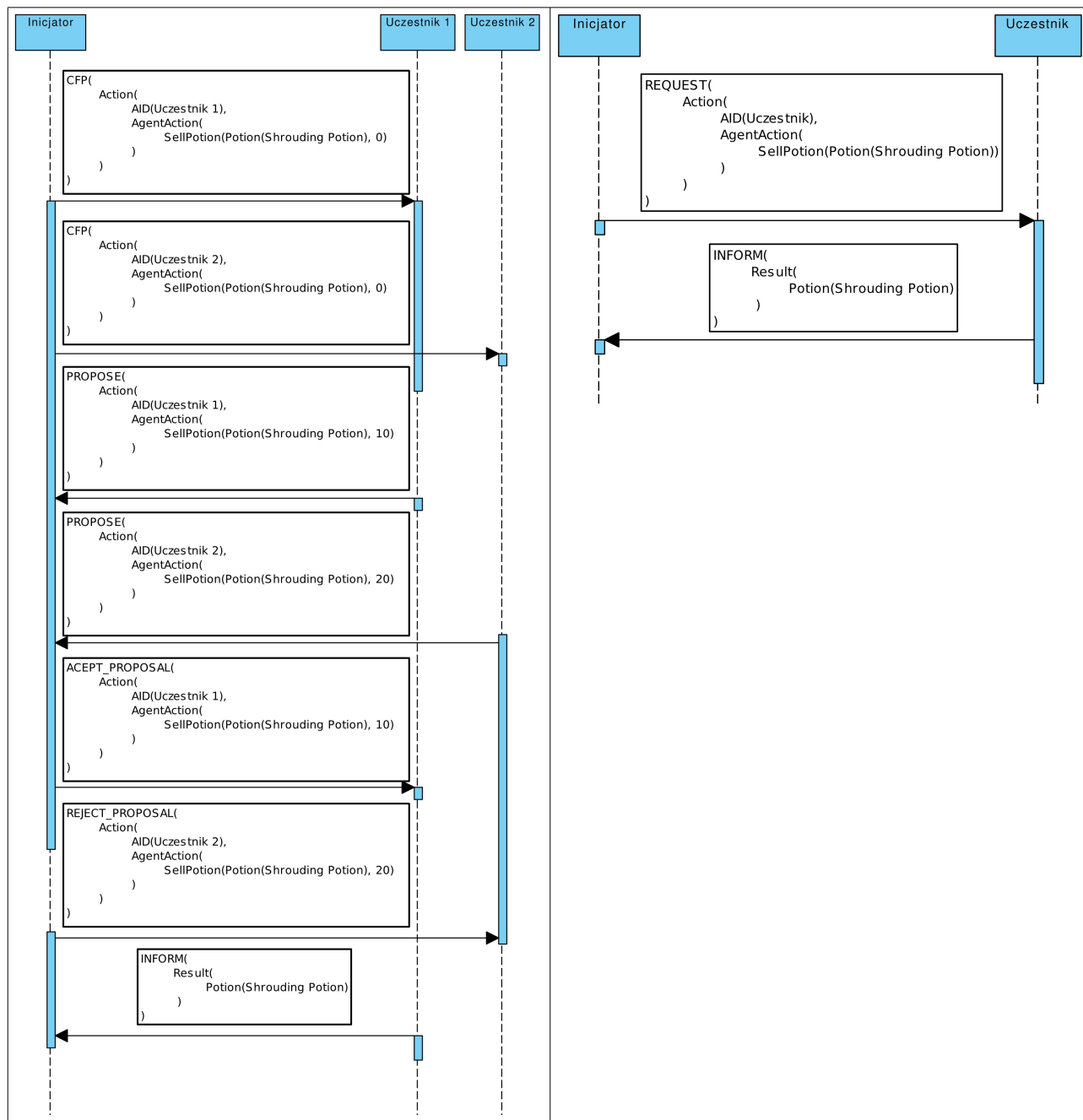
Opis interakcji

W oryginalnym przykładzie interakcja między agentami polegała na wysłaniu żądania przez agenta inicjującego i odpowiedzi przez agenta uczestniczącego. Agent inicjujący wysłał wiadomość typu `REQUEST` zawierającą opis akcji do wykonania (obiekt `Action` składający się z obiektów `AID` oraz obiektu implementującego interfejs `AgentAction`) a uczestnik odpowiadał wiadomością typu `REFUSE` gdy nie mógł wykonać akcji lub wiadomością `INFORM` zawierającą obiekt `Done` lub `Result` implementujące interfejs `Predicate`. Wiadomość żądania mogła być wysłana tylko do jednego agenta.

W ramach zadania należy przygotować samodzielnie klasy pozwalające na prowadzenie negocjacji. Protokół negocjacji zakłada że inicjator wysyła wiadomość typu `CFP` (Call For Proposals) zawierającą obiekt `Action` do więcej niż jednego agenta. Następnie uczestnicy odpowiadają wiadomością `REFUSE` gdy nie mogą zrealizować zadania lub wiadomością `PROPOSE` zawierającą obiekt `Action`. Obiekt implementujący `AgentAction` wysłany przez inicjatora zawiera jedynie opis akcji do wykonania, natomiast obiekt odesłany przez uczestników powinien zawierać konkretne warunki na jakich wykonają oni żadaną akcję. Przykładowo akcję przygotowania mikstur można zmodyfikować o koszt produkcji w złocie. Inicjator wyśle obiekt z wartością `cost` ustawioną na 0, natomiast uczestnicy odpowiedzą ustawiając konkretną wartość. Następnie gdy inicjator uzyska odpowiedzi od wszystkich agentów wybierze najkorzystniejszą propozycję i agentowi który ją zgłosił odeśle wiadomość typu `ACCEPT_PROPOSAL`, a wszystkim innym agentom `REJECT_PROPOSAL`. Agent który dostanie wiadomość `ACCEPT_PROPOSAL` odpowiada wiadomością `INFORM` zawierającą predykat `Done` lub `Result` (w zależności od tego czy akcja wymaga zwrócenia jakiegoś obiektu czy nie).

Przykłady komunikacji

Poniżej umieszczono przykład komunikacji zarówno dla przypadku żądanie-odpowiedź jak i prowadzenia negocjacji. Diagramy przedstawiają typy wiadomości oraz obiekty w nich przesyłane.



Zadania

Zadania do wykonania:

- przygotowania abstrakcyjnych klas służących prowadzeniu negocjacji,
- implementacja konkretnych zachowań na przykładzie sprzedaży mikstur,
- modyfikacja kodu alchemika tak aby lista mikstur i ich cena były przekazywane jako argument startowe agenta,
- modyfikacja kodu maga tak aby kupował najtańszą miksturę.

Uwagi

Zadanie nie zawiera dodatkowych modyfikacji. Zostaną one przedstawione jako elementy kolejnych zadań. Kod będzie wymagany w realizacji kolejnych zadań laboratoryjnych.