



Sprawozdanie

LABORATORIUM 6: PODSTAWY JADE

Kamil Udziela | RSI | Nr.indeksu: 286133

Przebieg ćwiczenia:

Ćwiczenie przedstawione w scenariuszu nr. 6 ma na celu zapoznanie nas z podstawą JADE oraz z korzystaniem z platform agentowych w języku JAVA.

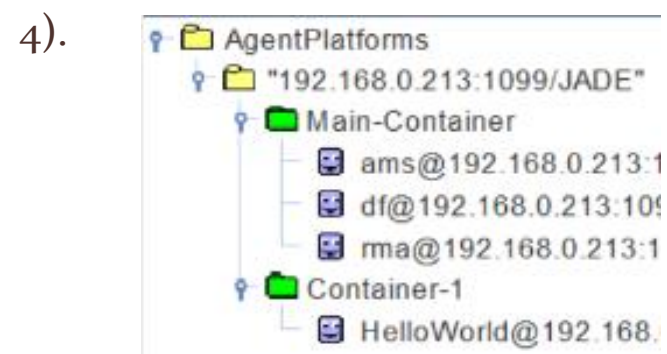
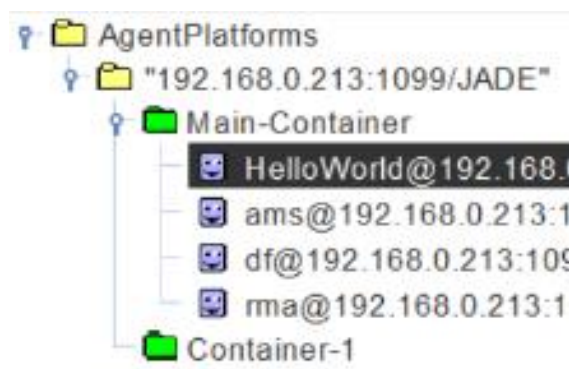
Etapy:

1). Pierwszym etapem ćwiczenia było uruchomienie platformy JADE oraz zapoznanie się z przykładowym programem (HelloWorldAgent). Jego uruchomienie oraz skompilowanie możliwe jest zarówno za pomocą GUI, jak i z linii komend.

2). Aby nasz agent nie usuwał się po wypisaniu tekstu na ekranie dokonałem modyfikacji klasy setup w pliku HelloWorldAgent.java. Usunąłem wywołanie metody `doDelete()` aby było to możliwe.

```
protected void setup(){  
    System.out.println("Hello World! My name is "+getLocalName());  
doDelete();  
}
```

3). W celu poznania pracy z platformą JADE, do uruchomionej platformy dodałem kolejny kontener. Nazwa kolejnego kontenera to „Container-1”. Kontener istniejący wcześniej – „Main-Container”



Następnie za pomocą polecenia **Migrate Agent** udało mi się przenieść jednego z agentów z głównego kontenera, do kontenera stworzonego przeze mnie w poprzednim podpunkcie. Przeniesiony agent to „HelloWorld@...”.

5). Użyłem polecenia **Clone Agent** aby sklonować przeniesionego wcześniej agenta.

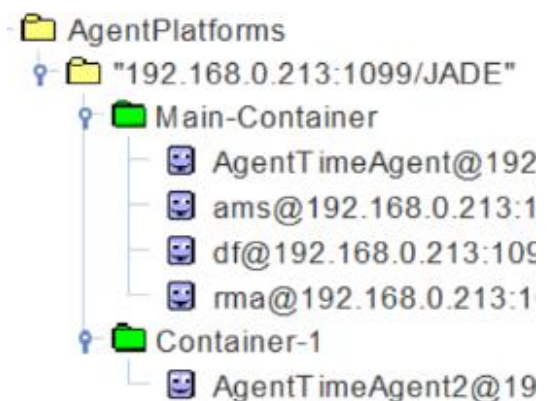


6). Usuwanie agenta odbywa się za pomocą polecenia **Kill**.

7). Następnie w przykładzie TimeAgent.java należało zmodyfikować czas, po którym agent się usuwał. Modyfikacja ta polegała na zmianie wartości, którą przyjmuje metoda WakerBehaviour. Początkowo była ustawiona na 10000, a więc 10 sekund. W poniższym przykładzie jest to miejsce zaznaczone na czerwono oraz pogrubione. Wartość finalna to 120000.

```
addBehaviour(new WakerBehaviour(this, 120000) {
    protected void handleElapsedTimeout() {
        System.out.println("Agent "+myAgent.getLocalName()+" : It's wakeup...
        myAgent.doDelete();
    }
});
```

8). Kolejnym zadaniem było przyłączenie kontenera do zdalnego hosta, a następnie dodanie klasy TimeAgent. Gdy to zrobiłem, otrzymałem dwóch agentów odliczających czas.



9). Zarówno skopiowanie agenta, jak i kopiowanie jego stanu sprawia, że oba agenty odliczają czas.

```
Wiersz polecenia - java jade.Boot -gui -container AgentTimeAgent2:T
Agent AgentTimeAgent2: tick=80
Agent AgentTimeAgent2@192.168.0.213:1099/JADE: tick=81
Agent AgentTimeAgent2: tick=81
```

10). Po dokonaniu migracji agenta z kontenera **Container** do kontenera **Main-Container** zauważyłem, że agent, który został przeniesiony zaczął odliczenie dopiero około 20 sekund po pierwszym agencie. Najprawdopodobniej jest to czas, który był potrzebny na przeniesienie agenta.

```
Wiersz polecenia - java jade.Boot -g
```

```
Agent AgentTimeAgent2: tick=1
```

```
Wiersz polecenia - java jade.Boot -
```

```
Agent AgentTimeAgent: tick=19
```

11). Próba stworzenia agenta w nieistniejącym kontenerze skończyła się błędem.