Artur Baruk  
Rafał Hazan

Dokumentacja projektu z przedmiotu Systemy Agentowe

Temat projektu: mobilni agenci w handlu elektronicznym.

# Rozwinięcie tematu projektu

Celem projektu jest zaprezentowanie wykorzystania środowiska wieloagentowego w handlu elektronicznym. System składa się z niezależnych sklepów (prowadzących handel detaliczny) oraz niezaleznych hurtowni (z których pochodzą produkty w sklepach). Zaopatrzeniem sklepów w towary zajmują się mobilni agenci, którzy działają na rynku niezależnie od sklepów i hurtownii. Agenci konkurują ze sobą, tak aby uzyskać jak najwyższe zyski.

# Sposób działania aktorów mobilnych

Agent odwiedzając sklep dowiaduje się jakich produktów brakuje oraz ceny po jakich może sprzedać dane produkty do sklepu. Odwiedzając hurtownie podejmuje decyzję, w której z nich dokonać zakupu zaopatrzenia. Poza tym agent nie może wydać więcej niż pozwala mu na to jego stan finansowy.

Na stan agenta składa się:

* Referencja (numer ID) obecnie obsługiwanego sklepu
* Referencja (numer ID) produktu na którego zostało zgłoszone zapotrzebowanie
* Tablicę aktualnych cen w sklepie

Agent aktualizuje swój stan będąc podpięty do sklepu.

Cykl działalności agenta:

1. Agent jest podpięty do sklepu, otrzymuje zamówienie na dany produkt oraz aktualny cennik sklepu.
2. Agent zapamiętuje szczegóły zamówienia i podpina się do pierwszej hurtownii.
3. Podejmuje decyzje, czy opłaca mu się kupić towar. Jeśli tak, to dokonuje zakupu i zmniejsza stan posiadanych pieniędzy. Dostarcza produkt do sklepu, który złożył zamówienie i zwiększa stan posiadanych pieniędzy. Tym samym kończy obecny cykl i zaczyna nowy.
4. W przeciwnym wypadku przepina się kolejno do pozostałych hurtownii, aż znajdzie korzystną ofertę (jeśli nie to dokonuje zakupu w ostatniej dostępnej). Dokonuje zakupu i zmniejsza stan posiadanych pieniędzy. Dostarcza produkt do sklepu, który złożył zamówienie i zwiększa stan posiadanych pieniędzy. Tym samym kończy obecny cykl i zaczyna nowy.

# Struktura projektu

Projekt został zaimplementowany w jęzuku Scala z użyciem biblioteki Akka. Wybór tej platformy zapewnia zwięzły i bezpieczny sposób komunikacji w aplikacjach wielowątkowych. Każdy sklep lub hurtownia działa na innym węźle. Każdy węzeł może być utworzony pod innym adresem IP. Komunikacja (przenoszenie agentów) między węzłami zachodzi przy pomocy protokołu TCP, co zapewnia dostarczenie niezmodyfikowanej wiadomości. Poniżej przedstawiona jest struktura projektu. Przerywane strzałki oznaczają asynchroniczną komunikację.

# D:\Studies\Semestr_10\sag\projekt\sag\docs\struktura systemu.png

* Producent zdarzeń dostawy produktu. Odpowiada za symulację dostaw do pojedynczego magazynu w czasie rzeczywistym. Działa asynchronicznie na węźle hurtowni.
* Producent zdarzeń braku produktu. Odpowiada za symulację tworzenia braków produktów w pojedynczym sklepie w czasie rzeczywistym. Działa asynchronicznie na węźle sklepu.
* Przedstawiciel hurtowni. Aktor pozwalający na komunikację agentów z hurtownią. Pośrednik zakupów. Posiada aktualny stan magazynowy hurtowni oraz jej cennik. Działa na pojedynczym węźle.
* Przedstawiciel sklepu. Aktor pozwalający na komunikację agentów ze sklepem. Pośrednik zakupów. Posiada swoich agentów, którzy dokonują dla niego zakupów. Zawiera aktualny cennik produktów. Działa na pojedynczym węźle.
* Agent. Atomowa komórka poszukująca produktów po jak najlepszej cenie. Komunikuje się asynchronicznie ze swoim sklepem oraz hurtowniami. Posiada stan, który jest przenoszony między węzłami. Dzięki niemu wie jakie akcje wykonać. W stanie agenta zawarte są takie informacje jak, id aktualnego węzła, informacje o poszukiwanym produkcie, wielkość portfela.

# Uruchomienie programu

Razem ze źródłami, zostały przesłane dwie paczki JAR. Jedna odpowiada za aplikację jednej hurtowni, a druga uruchamia aplikację jednego sklepu z agentami. Chcąc uruchomić symulację należy uruchomić wymaganą liczbę hurtowni i sklepów z odpowiednimi identyfikatorami. Pojedynczą aplikację uruchamia się standardowo, z podaniem id sklepu/hurtowni jako argumentem uruchomienia:

java –jar store.jar 0

Dla zdefiniowanej domyślnie symulacji należy uruchomić dwa sklepy (identyfikatory od 0 i 1) i dwie hurtownie (identyfikatory od 0 do 1). Chcąc zmienić parametry lub zmienić liczbę poszczególnych węzłów należy zmienić konfigurację Application.conf programu i skompilować ten projekt na nowo przy pomocy wtyczki do narzędzia sbt o nazwie one-jar.