

J2EE - projekt zaliczeniowy

Magdalena Dukielska
md209461@students.mimuw.edu.pl

Paweł Banasik
pb209450@students.mimuw.edu.pl

1 marca 2007

1 Cel projektu

Założeniem projektu jest stworzenie darmowego i prostego w użyciu środowiska do pisania programów korzystających z technologii Java Spaces. Pozwoliłoby to na upowszechnienie się Java Spaces, ponieważ główną przeszkodą w tworzeniu oprogramowania działającego w tej technologii jest skomplikowany proces konfiguracji środowiska, w którym można uruchamiać pisane programy.

Podstawowa funkcjonalność, którą chcemy zrealizować to stworzenie wtyczki do Eclipse/Netbeans IDE o następujących możliwościach:

- tworzenie projektu Java Spaces (od razu ustawione odpowiednie biblioteki, w tym biblioteka z kodem do wyszukiwania przestrzeni krotek na podstawie nazwy - jej napisanie jest częścią naszego zadania, istniejące w tej chwili rozwiązanie wymaga korzystania z trudnego w zrozumieniu mechanizmu kontenera Jini) - stworzona w ten sposób konfiguracja wymagałaby od użytkownika wtyczki napisania jedynie samego kodu procesów, które mają korzystać z przestrzeni krotek
- uruchamianie programów - nie powinno wymagać od użytkownika wpisywania żadnych dodatkowych opcji ani podawania odpowiednich argumentów dla JVM (może stworzyć anotacje, które zmieniałyby domyślne ustawienia?)
- zarządzanie kontenerem Jini obsługującym m.in. Java Spaces - skoncentrujemy się tylko na funkcjonalności bezpośrednio związanej z Java Spaces, tzn. wtyczka ma pozwalać na uruchamianie, zatrzymywanie kontenera, przeglądanie istniejących przestrzeni krotek, ich usuwanie, czyszczenie z krotek a także tworzenie nowych przestrzeni krotek
- znajdowanie przestrzeni krotek o danej nazwie - mechanizm dependency injection z anotacjami wskazującymi interesującą nas nazwę przestrzeni krotek
- przeglądanie zawartości wybranej przestrzeni krotek - prosty język zapytań o krotki znajdujące się w przestrzeni oparty na wzorcach (tak samo jak w Lindzie), możliwość usuwania i dodawania określonych krotek
- stworzenie alternatywnego widoku kodu programu - analogicznego do składni Lindy z Programowania Współbieżnego - zapis sygnatury krotki, wzorce krotek do dopasowania (przykłady składni są na waźniaku)
- automatyczne generowanie klasy implementującej interfejs Entry dla zadanej sygnatury krotki
- możliwość obserwowania działania napisanych przez użytkownika procesów działających na wskazanej przestrzeni krotek. Przykładowo użytkownik mógłby zatrzymać w którymś miejscu wybrany proces, ewentualnie zmienić zawartość przestrzeni krotek a potem wznowić działanie procesu. Przydatna byłaby też wizualizacja współdziałania procesów. Taką funkcjonalność można by było zaimplementować za pomocą aspektów.