

# Možnosti rozširovania programovacieho jazyka Java



TECHNICKÁ  
UNIVERZITA  
V KOŠICIACH

Autor: Ing. Viktor Dobránsky

Vedúci práce: prof. Ing. Ján Kollár CSc.

## Cieľ

Implementovať doménovo-špecifický jazyk pomocou rozšírenia syntaxe jazyka *Java*.

## Inšpirácia

Hlavná idea vytvoreného rozšírenia bola čerpaná z nástroja *LINQ* určeného pre jazyky *C#* a *Visual Basic*. Tento nástroj umožňuje písať dopyty na selekciu údajov priamo do zdrojového kódu. Tieto dopyty používajú kľúčové slová zo štandardu *SQL*. Nástroj *LINQ* umožňuje selekciu údajov z databáz, kolekcii, *XML* súborov, atď.

## Postup

Na tvorbu rozšírenia bola použitá sada nástrojov *Ahead Tool Suite*. Tá je priamo určená pre tvorbu jazykov. Na definíciu syntaxe bol použitý jazyk *Bali* a na definovanie sémantiky jazyk *Jakarta*.

## Výsledok

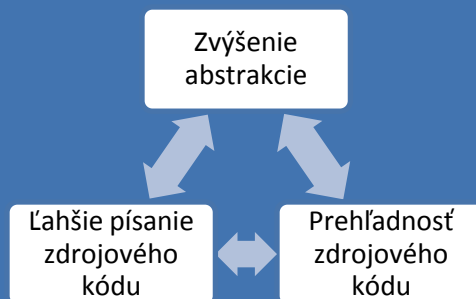
Vytvorené rozšírenie umožňuje selekciu dát iba nad kolekciami a manipuláciu s údajmi zoskupovaním, zoraďovaním, aplikovaním podmienok, výberom viacerých údajov a pod. Výsledkom práce je program, ktorý je schopný preložiť zdrojový kód s rozšírením do zdrojového kódu v čistej *Java*, ktorý môže byť následne preložený *javac* kompilátorom.

## Referencie

Ahead Tool Suite website [online]. 2008. Dostupné na internete: <<http://www.cs.utexas.edu/~schwartz/ATS/fopdocs>>.

PIALORSI, Paolo - RUSSO, Marco: *Introducing Microsoft LINQ*. Redmond: Microsoft Press, 2007. 203 s. BPN X13-68390.

SARVELA, Jacob Neal: *The Bali Language* [pdf]. 2003. Dostupné na internete: <<http://www.cs.utexas.edu/~schwartz/ATS/fopdocs/bali.pdf>>.



## Ukážka použitia rozšírenia

```
public void SomeMethod(){
    // kolekcia naplnená údajmi
    List<Car> cars = new ArrayList<Car>();
    cars.add(new Car("1234", "Skoda", "Yellow"));
    /*
     * nasledujúci príkaz vytvorí zoznam s názvom idC,
     * vyberajú sa identifikačné čísla a farby jednotlivých
     * áut
     */
    var idC = FROM c IN cars
              SELECT new {c.getId(), c.getColor()};

    // komplikovanejší príklad je stále ľahko čitateľný
    var foo = FROM c IN cars
              WHERE c.getId().startsWith("1")
              GROUP c BY c.getName() INTO g
              ORDERBY c.getName() DESCENDING
              SELECT new {
                Name = g.key,
                Cars = g
              };

    System.out.println(idC);
    System.out.println(foo);
    // viac príkladov nájdete TU
}
```